Politecnico di Torino

Politecnico di Torino

Corso di Laurea Magistrale in Ingegneria Aerospaziale
Orientamento Aerostrutture
A.a. 2022/2023
Sessione di Laurea Aprile 2023

Sviluppo di modelli global/local per analisi FEM mediante reti neurali

Relatori:
Marco Petrolo
Alfonso Pagani
Pierluigi Iannotti
Mattia Trombini

Candidato:
Mattia Melis

A mio nonno Agostino

Sommario

Nell'ambito dell'ingegneria aerospaziale viene richiesta la massima precisione e attenzione ai dettagli, trattandosi di progettazione e sviluppo di sistemi molto complessi. L'analisi strutturale è una sua branchia che si pone l'obiettivo di studiare la sicurezza ed affidabilità delle strutture nella loro totalità, andando ad individuare, prevenire e correggere eventuali criticità e malfunzionamenti che potrebbero verificarsi. Il progresso tecnologico degli ultimi decenni ha portato notevoli miglioramenti in questo ambito, grazie allo sviluppo di software di calcolo in grado di simulare e modellizzare al computer, con sempre più precisione, le strutture reali. Inoltre, sono sempre più numerose le funzioni ed i tipi di analisi che questi sono in grado di svolgere nelle molteplici casistiche di geometrie, condizioni di carico, condizioni di vincolo e tipo di materiali, per citarne alcune. Questo processo di analisi strutturale preliminare risulta essere di estrema importanza, ancor di più nei casi in cui si voglia sviluppare un sistema che durante la sua vita operativa preveda la presenza di persone quali equipaggio e possibili passeggeri, di cui la sicurezza è fondamentale.
Tuttavia, una maggiore affidabilità la si ottiene utilizzando codici basati su modelli di calcolo molto avanzati e complessi, che vanno ad operare su modelli di strutture complesse a loro volta. Questo implica come aspetto negativo un costo computazionale sempre più oneroso, che si traduce nella necessità di possedere risorse informatiche sempre più avanzate per poter fornire la capacità di elaborazione necessaria. Oltre a ciò, un costo computazionale sempre più elevato implica anche che un'analisi strutturale avanzata possa richiedere un tempo di calcolo di molte ore o, addirittura, giorni. In questa tesi si vuole quindi andare ad affrontare questa sfida molto attuale con l'obiettivo di ridurre il costo computazionale ed il tempo necessario richiesto al calcolatore, proponendo l'introduzione e lo sviluppo di reti neurali artificiali adeguate da poter essere utilizzate come modelli surrogati per l'ottenimento di risultati di natura strutturale.
Lo svolgimento è iniziato con l'introduzione della Carrera Unified Formulation (CUF), che grazie alla sua notazione inidiciale ed alle altre sue potenzialità, è in grado di ridurre già in parte il costo computazionale mediante l'utilizzo di un software di calcolo apposito per l'analisi strutturale ad elementi finiti. Dopo averne verificato la validità, viene effettuato un primo studio riguardante il machine learning e le reti neurali artificiali, prestando particolare attenzione al tipo di architetture di reti possibili e a come i vari parametri vadano ad influenzare il loro funzionamento.
Il vantaggio che se ne vuole trarre è quello di sfruttare la capacità della rete neurale di predire il comportamento della struttura sottoposta a differenti carichi e condizioni senza dover effettuare delle simulazioni dettagliate del problema. Per riuscire a fare ciò è necessaria comunque una particolare attenzione sulla quantità e qualità dei dati da fornire alla rete, tramite i quali può apprendere le complesse relazioni intrinseche, così come è necessario un accurato controllo sull'affidabilità dei risultati riprodotti.
La rete neurale che si è sviluppata è stata applicata nell'ottenimento di Best Theory Diagrams (BTD), da poter confrontare con quelli ottenibili tramite il codice di risoluzione strutturale. Inizialmente, l'ambito di funzionamento è stato quello delle analisi a

vibrazioni libere, riproducendo le prime frequenze naturali a cui vibra il sistema. Successivamente, dopo aver dimostrato le potenzialità di questo approccio e dell'utilizzo della rete neurale artificiale, si è voluto applicare lo stesso metodo per analisi di tipo global/local andando a confrontare le distribuzioni di teorie migliori sulla struttura predette dalla rete, con quelle validate dal codice di calcolo. Infine, è stata investigata un'ulteriore applicazione riguardante la previsione dei valori di failure index e delle tensioni in precisi punti della struttura.

Indice

Sommario **5**

Elenco delle Figure **10**

Elenco delle Tabelle **16**

1 Introduzione **19**
1.1 Machine learning ed analisi strutturali 22
1.2 Struttura dell'elaborato 23

2 Modelli strutturali **25**
2.1 Principio degli spostamenti virtuali 25
2.1.1 Analisi statica 25
2.1.2 Analisi vibrazioni libere 26
2.1.3 Analisi dinamica 27
2.2 Carrera Unified Formulation 27
2.2.1 Elemento Barra 27
2.2.2 Tecnica di assemblaggio CUF 30
2.2.3 Modelli 1D, 2D e 3D 31
2.2.4 Modello trave 32
2.2.5 Modello piastra e guscio 34
2.3 Funzioni di espansione 37
2.3.1 Espansioni di Taylor 37
2.3.2 Espansioni di Lagrange 38
2.4 Global/Local 41
2.5 Metodo assiomatico-asintotico 42
2.5.1 Best Theory Diagram 42
2.5.2 Best Theory Distribution 43

3 Reti neurali **45**
3.1 Funzioni di attivazione 49
3.1.1 Rectified Linear Unit 49
3.1.2 Sigmoid 50
3.2 Training della rete 51
3.3 Tipologia di layer 53
3.3.1 Fully connected 53
3.3.2 Convolutional 53
3.3.3 Flatten 54

4 Risultati preliminari **55**
4.1 Analisi meccaniche . . . 55
4.1.1 Analisi statiche trave . . . 55
4.1.2 Replica risultati paper . . . 58
4.1.3 Analisi vibrazioni libere . . . 63
4.2 Testing reti neurali . . . 64
4.2.1 Best theory diagrams . . . 64
4.2.2 Reti neurali . . . 66

5 Sviluppo rete neurale **68**
5.1 Interfaccia tra rete neurale e software strutturale . . . 68
5.1.1 Utilizzo preliminare del software strutturale . . . 68
5.1.2 Utilizzo della rete neurale . . . 69
5.2 Rete neurale multi-output . . . 71
5.2.1 Architettura rete . . . 72
5.2.2 Variazione dei parametri . . . 73
5.2.3 Risultati . . . 75
5.3 Analisi vibrazioni libere . . . 84
5.3.1 Semplice appoggio 0/90/0 . . . 84
5.3.2 Clamped - Free 0/90/0 . . . 95

6 Analisi Global/Local **103**
6.1 S-S 0/90/0 a/h=10 . . . 104
6.1.1 Media . . . 104
6.1.2 10 Frequenze . . . 106
6.2 S-S 0/90/0 a/h=10 - Dati training ridotti . . . 111
6.2.1 Media . . . 112
6.2.2 10 Frequenze . . . 113
6.3 S-S 0/90/0 a/h=100 . . . 119
6.3.1 Media . . . 119
6.3.2 10 Frequenze . . . 121
6.4 C-F 0/90/0 a/h=10 . . . 126
6.4.1 Media . . . 126
6.4.2 10 Frequenze . . . 128
6.5 Riepilogo . . . 133

7 Analisi Failure index **137**
7.1 Output multiplo . . . 139
7.2 Output singolo . . . 141
7.3 Variazione dell'output . . . 147

8 Conclusioni **149**
8.1 Prospettive future . . . 150

A Grafici BTD **151**

B Grafici Global/Local **158**

Bibliografia **173**

Ringraziamenti **176**

Elenco delle figure

2.1 Elemento barra a 2 nodi. 29
2.2 Schema assemblaggio CUF. 31
2.3 Modello trave. 32
2.4 Modello Piastra. 34
2.5 Modello Guscio. 36
2.6 Normalizzazione elemento quadrangolare a quattro nodi (L4) tramite polinomi di Lagrange. 39
2.7 Normalizzazione elemento quadrangolare a nove nodi (L9) tramite polinomi di Lagrange. 40
2.8 Rappresentazione Best Theory Distribution. 44

3.1 Schema funzionamento neurone biologico. 45
3.2 Schema funzionamento ANN. 46
3.3 Schema neurone a singolo input. 47
3.4 Schema neurone ad input multiplo. 48
3.5 Schema layer di S neuroni. 48
3.6 Schema rete neurale a due layer. 49
3.7 Funzione ReLU. 50
3.8 Funzione Sigmoid. 51
3.9 Operazione di convoluzione. 54

4.1 Rappresentazione geometria della trave. 55
4.2 Elementi 1D. 56
4.3 Elementi 2D. 56
4.4 Deformata con diverse discretizzazioni. 57
4.5 Sezione rettangolare. 58
4.6 Sezione a C. 60
4.7 Discretizzazioni sezione a C. 61
4.8 Sezione quadrata cava aperta. 62
4.9 Discretizzazioni sezione quadrata cava aperta. 62
4.13 Influenza parametro epochs. 67
4.14 Influenza parametro batch size. 67

5.1 Ilustrazione metodologia applicata. 68
5.2 Schema di utilizzo del software strutturale. 69
5.3 Schema di utilizzo della rete neurale. 71
5.4 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze I e II. 86

5.5 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze III e IV. 87
5.6 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenza V. 87
5.7 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5 - Frequenze I e II. 87
5.8 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5 - Frequenze III e IV. 88
5.9 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5 - Frequenza V. 88
5.10 Piastra semplicemente appoggiata, 0/90/0, a/h=10 - Frequenze I e II. . 89
5.11 Piastra semplicemente appoggiata, 0/90/0, a/h=10 - Frequenze III e IV. 89
5.12 Piastra semplicemente appoggiata, 0/90/0, a/h=10 - Frequenza V. . . . 89
5.13 Piastra clamped - free, 0/90/0, a/h=4, R/a=10 - Frequenze I e II. . . . 97
5.14 Piastra clamped - free, 0/90/0, a/h=4, R/a=10 - Frequenze III e IV. . 97
5.15 Piastra clamped - free, 0/90/0, a/h=4, R/a=10 - Frequenza V. 98
5.16 Piastra clamped - free, 0/90/0, a/h=20, R/a=10 - Frequenze I e II. . . 98
5.17 Piastra clamped - free, 0/90/0, a/h=20, R/a=10 - Frequenze III e IV. . 98
5.18 Piastra clamped - free, 0/90/0, a/h=20, R/a=10 - Frequenza V. 99

6.1 Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 104
6.2 Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 105
6.3 Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 106
6.4 Frequenza I - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 106
6.5 Frequenza I - Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 107
6.6 Frequenza I - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 107
6.7 Frequenza II - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 107
6.8 Frequenza II - Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 108
6.9 Frequenza II - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 108
6.10 Frequenza III - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 108
6.11 Frequenza III - Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 109
6.12 Frequenza III - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 109
6.13 Frequenza IV - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 109
6.14 Frequenza IV - Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 110
6.15 Frequenza IV - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 110
6.16 Frequenza V - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 110
6.17 Frequenza V - Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 111
6.18 Frequenza V - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 111
6.19 Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 112
6.20 Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 113
6.21 Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 113
6.22 Frequenza I - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 114
6.23 Frequenza I - Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 114
6.24 Frequenza I - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 114

6.25 Frequenza II - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 115
6.26 Frequenza II - Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 115
6.27 Frequenza II - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 115
6.28 Frequenza III - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 116
6.29 Frequenza III - Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 116
6.30 Frequenza III - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 116
6.31 Frequenza IV - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 117
6.32 Frequenza IV - Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 117
6.33 Frequenza IV - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 117
6.34 Frequenza V - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 118
6.35 Frequenza V - Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 118
6.36 Frequenza V - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 118
6.37 Quattro N4, S-S, 0/90/0, a/h=100, R/a=5. 120
6.38 Otto N4, S-S, 0/90/0, a/h=100, R/a=5. 120
6.39 Dodici N4, S-S, 0/90/0, a/h=100, R/a=5. 120
6.40 Frequenza I - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5. 121
6.41 Frequenza I - Otto N4, S-S, 0/90/0, a/h=100, R/a=5. 121
6.42 Frequenza I - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5. 122
6.43 Frequenza II - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5. 122
6.44 Frequenza II - Otto N4, S-S, 0/90/0, a/h=100, R/a=5. 122
6.45 Frequenza II - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5. 123
6.46 Frequenza III - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5. 123
6.47 Frequenza III - Otto N4, S-S, 0/90/0, a/h=100, R/a=5. 123
6.48 Frequenza III - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5. 124
6.49 Frequenza IV - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5. 124
6.50 Frequenza IV - Otto N4, S-S, 0/90/0, a/h=100, R/a=5. 124
6.51 Frequenza IV - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5. 125
6.52 Frequenza V - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5. 125
6.53 Frequenza V - Otto N4, S-S, 0/90/0, a/h=100, R/a=5. 125
6.54 Frequenza V - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5. 126
6.55 Quattro N4, C-F, 0/90/0, a/h=10, R/a=5. 126
6.56 Otto N4, C-F, 0/90/0, a/h=10, R/a=5. 127
6.57 Dodici N4, C-F, 0/90/0, a/h=10, R/a=5. 128
6.58 Frequenza I - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5. 128
6.59 Frequenza I - Otto N4, C-F, 0/90/0, a/h=10, R/a=5. 129
6.60 Frequenza I - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5. 129
6.61 Frequenza II - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5. 129
6.62 Frequenza II - Otto N4, C-F, 0/90/0, a/h=10, R/a=5. 130
6.63 Frequenza II - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5. 130
6.64 Frequenza III - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5. 130
6.65 Frequenza III - Otto N4, C-F, 0/90/0, a/h=10, R/a=5. 131
6.66 Frequenza III - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5. 131
6.67 Frequenza IV - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5. 131
6.68 Frequenza IV - Otto N4, C-F, 0/90/0, a/h=10, R/a=5. 132

6.69 Frequenza IV - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5. 132
6.70 Frequenza V - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5. 132
6.71 Frequenza V - Otto N4, C-F, 0/90/0, a/h=10, R/a=5. 133
6.72 Frequenza V - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5. 133
6.73 Frequenza I. 134
6.74 Frequenza II. 134
6.75 Frequenza III. 134
6.76 Frequenza IV. 135
6.77 Frequenza V. 135
6.78 Frequenza VI. 135
6.79 Frequenza VII. 135
6.80 Frequenza VIII. 136
6.81 Frequenza IX. 136
6.82 Frequenza X. 136

7.1 Confronto FEM - NN, caso a 5DOF 25 punti ed 81 punti. 145
7.2 Confronto FEM - NN, caso a 8DOF 25 punti ed 81 punti. 145
7.3 Confronto FEM - NN, caso a 12DOF 25 punti ed 81 punti. 146

A.1 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenza VI. 151
A.2 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze VII e VIII. 151
A.3 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze IX e X. 152
A.4 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze XI e XII. 152
A.5 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze XIII e XIV. 152
A.6 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze XV e XVI. 153
A.7 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze XVII e XVIII. 153
A.8 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze XIX e XX. 153
A.9 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5 - Frequenza VI. 154
A.10 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5 - Frequenze VII e VIII. 154
A.11 Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5 - Frequenze IX e X. 154
A.12 Piastra semplicemente appoggiata, 0/90/0, a/h=10 - Frequenza VI. . . 155
A.13 Piastra semplicemente appoggiata, 0/90/0, a/h=10 - Frequenze VII e VIII. 155

A.14 Piastra semplicemente appoggiata, 0/90/0, a/h=10 - Frequenze IX e X. 155
A.15 Piastra clamped - free, 0/90/0, a/h=4, R/a=10 - Frequenza VI. 156
A.16 Piastra clamped - free, 0/90/0, a/h=4, R/a=10 - Frequenze VII e VIII. 156
A.17 Piastra clamped - free, 0/90/0, a/h=4, R/a=10 - Frequenze IX e X. . . 156
A.18 Piastra clamped - free, 0/90/0, a/h=20, R/a=10 - Frequenza VI. 157
A.19 Piastra clamped - free, 0/90/0, a/h=20, R/a=10 - Frequenze VII e VIII. 157
A.20 Piastra clamped - free, 0/90/0, a/h=20, R/a=10 - Frequenze IX e X. . 157

B.1 Frequenza VI - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 158
B.2 Frequenza VI - Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 158
B.3 Frequenza VI - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 159
B.4 Frequenza VII - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 159
B.5 Frequenza VII - Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 159
B.6 Frequenza VII - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 160
B.7 Frequenza VIII - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 160
B.8 Frequenza VIII - Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 160
B.9 Frequenza VIII - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 161
B.10 Frequenza IX - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 161
B.11 Frequenza IX - Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 161
B.12 Frequenza IX - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 162
B.13 Frequenza X - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5. 162
B.14 Frequenza X - Otto N4, S-S, 0/90/0, a/h=10, R/a=5. 162
B.15 Frequenza X - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5. 163
B.16 Frequenza VI - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5. 163
B.17 Frequenza VI - Otto N4, S-S, 0/90/0, a/h=100, R/a=5. 163
B.18 Frequenza VI - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5. 164
B.19 Frequenza VII - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5. 164
B.20 Frequenza VII - Otto N4, S-S, 0/90/0, a/h=100, R/a=5. 164
B.21 Frequenza VII - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5. 165
B.22 Frequenza IX - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5. 165
B.23 Frequenza IX - Otto N4, S-S, 0/90/0, a/h=100, R/a=5. 165
B.24 Frequenza IX - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5. 166
B.25 Frequenza X - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5. 166
B.26 Frequenza X - Otto N4, S-S, 0/90/0, a/h=100, R/a=5. 166
B.27 Frequenza X - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5. 167
B.28 Frequenza VI - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5. 167
B.29 Frequenza VI - Otto N4, C-F, 0/90/0, a/h=10, R/a=5. 167
B.30 Frequenza VI - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5. 168
B.31 Frequenza VII - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5. 168
B.32 Frequenza VII - Otto N4, C-F, 0/90/0, a/h=10, R/a=5. 168
B.33 Frequenza VII - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5. 169
B.34 Frequenza VIII - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5. 169
B.35 Frequenza VIII - Otto N4, C-F, 0/90/0, a/h=10, R/a=5. 169
B.36 Frequenza VIII - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5. 170

B.37 Frequenza IX - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5. 170
B.38 Frequenza IX - Otto N4, C-F, 0/90/0, a/h=10, R/a=5. 170
B.39 Frequenza IX - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5. 171
B.40 Frequenza X - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5. 171
B.41 Frequenza X - Otto N4, C-F, 0/90/0, a/h=10, R/a=5. 171
B.42 Frequenza X - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5. 172

Elenco delle tabelle

2.1 Polinomi espansione di Taylor modello 1D. 37
2.2 Polinomi espansione di Taylor modello 2D. 38
2.3 Coordinate normalizzate dei nodi elemento L4. 40
2.4 Coordinate normalizzate dei nodi elemento L9. 41
2.5 Rappresentazione posizione dei termini attivi. 43

4.1 Spostamento verticale u_z del punto $[0, L/2, 0]$. 57
4.2 Spostamento verticale u_z trave a sezione rettangolare. $L/h = 100$. . . . 59
4.3 Spostamento verticale u_z trave a sezione rettangolare. $L/h = 10$. 59
4.4 Effetto di diverse discretizzazioni sullo spostamento verticale u_z trave a sezione rettangolare. $L/h = 10$. 60
4.5 Spostamento verticale u_z trave sezione a C. 61
4.6 Spostamento u_x trave sezione quadrata cava aperta. 63
4.7 Valori delle prime dieci frequenze. 63
4.8 Output script Python - BTD. 65

5.1 Distribuzione percentuale scelta dati di input. 70
5.2 Architettura rete neurale - output multidimensionale. 73
5.3 Parametri finali rete neurale artificiale multi-output. 74
5.4 Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza I. 76
5.5 Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza II. 76
5.6 Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza III. 77
5.7 Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza IV. 77
5.8 Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza V. 78
5.9 Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza VI. 78
5.10 Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza VII. 79
5.11 Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza VIII. 79
5.12 Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza IX. 80
5.13 Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza X. 80
5.14 Semplice appoggio, 0/90, a/h=100, R/a=20 - Media Frequenze. 81
5.15 Disposizione dei termini. 82
5.16 Legenda colori. 82
5.17 BTD complessivo 5 DOF - Caso **D**. 82
5.18 BTD complessivo 8 DOF - Caso **D**. 83
5.19 BTD complessivo 12 DOF - Caso **D**. 83
5.20 0/90/0, a/h=10, $\overline{\omega} = \omega\sqrt{\frac{\rho a^4}{h^2 E_T}}$. 84
5.21 BTD 0/90/0, a/h=10, R/a=2. 85

5.22 BTD 0/90/0, a/h=10, R/a=5. 85
5.23 BTD 0/90/0, a/h=10, plate. 86
5.24 BTD complessivo 5 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2. 90
5.25 BTD complessivo 8 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2. 91
5.26 BTD complessivo 12 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2. 92
5.27 BTD complessivo 5 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5. 93
5.28 BTD complessivo 8 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5 93
5.29 BTD complessivo 12 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5. 94
5.30 BTD complessivo 5 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10. 94
5.31 BTD complessivo 8 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10. 95
5.32 BTD complessivo 12 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10. 95
5.33 0/90/0, R/a=10, $\overline{\omega} = \omega\sqrt{\frac{\rho a^4}{h^2 E_T}}$. 96
5.34 BTD 0/90/0, R/a=10, a/h=4. 96
5.35 BTD 0/90/0, R/a=10, a/h=20. 96
5.36 BTD complessivo 5 DOF - Piastra clamped/free, 0/90/0, R/a=10, a/h=4. 99
5.37 BTD complessivo 8 DOF - Piastra clamped/free, 0/90/0, R/a=10, a/h=4.100
5.38 BTD complessivo 12 DOF - Piastra clamped/free, 0/90/0, R/a=10, a/h=4. 100
5.39 BTD complessivo 5 DOF - Piastra clamped/free, 0/90/0, R/a=10, a/h=20. 101
5.40 BTD complessivo 8 DOF - Piastra clamped/free, 0/90/0, R/a=10, a/h=20. 101
5.41 BTD complessivo 12 DOF - Piastra clamped/free, 0/90/0, R/a=10, a/h=20. 102

6.1 S-S, 0/90/0, a/h=10, R/a=5, $\overline{\omega} = \omega\sqrt{\frac{\rho a^4}{h^2 E_T}}$. 104
6.2 S-S, 0/90/0, a/h=10, R/a=5. 105
6.3 S-S, 0/90/0, a/h=10, R/a=5, 10% dati input nn. 112
6.4 S-S, 0/90/0, a/h=100, R/a=5. 119
6.5 C-F, 0/90/0, a/h=10, R/a=5. 127

7.1 Coordinate dei punti. a/h=10. 139
7.2 Coordinate dei punti. a/h=100. 139
7.3 Semplice appoggio, 0/90/0, R/a=5, a/h=10 - Valori analisi statica. . . 140

7.4 Semplice appoggio, 0/90/0, R/a=5, a/h=10 - Valori rete neurale. . . . 140
7.5 Semplice appoggio, 0/90/0, R/a=5, a/h=100 - Valori analisi statica. . . 141
7.6 Semplice appoggio, 0/90/0, R/a=5, a/h=100 - Valori rete neurale. . . . 141
7.7 Coordinate dei punti. a/h=100. 142
7.8 Semplice appoggio, 0/90/0, R/a=5, a/h=100 - Valori analisi statica. . . 142
7.9 Semplice appoggio, 0/90/0, R/a=5, a/h=100 - Valori rete neurale. . . . 143
7.10 BTD Punto 1 (0.25, 0.25, -0.005). 143
7.11 BTD Punto 3 (0.5, 0.5, -0.005). 144
7.12 BTD 5 DOF - Confronto analisi FEM - rete neurale a 3, 25, 81 punti. . 146
7.13 BTD 8 DOF - Confronto analisi FEM - rete neurale a 3, 25, 81 punti. . 146
7.14 BTD 12 DOF - Confronto analisi FEM - rete neurale a 3, 25, 81 punti. 147
7.15 Confronto prediction tensioni su due punti. 147
7.16 Confronto prediction tensioni normalizzate su due punti. 148

Capitolo 1

Introduzione

Con il termine *Machine Learning* (ML) si intende quella branca dell'intelligenza artificiale il cui scopo è lo sviluppo di codici ed algoritmi che permettano ai computer di ottenere la capacità di apprendere da dati forniti, senza essere stati esplicitamente programmati in ogni operazione necessaria. Gli studi appartenenti a questa branca dell'informatica sono stati sviluppati a partire dalla seconda metà del ventesimo secolo, in cui è possibile individuare un primo elaborato molto interessante di Arthur Samuel, colui che coniò il termine di machine learning, prodotto nel 1959. In [1] vengono analizzate due differenti procedure di sviluppo per il machine learning, sfruttando il gioco della dama. Il primo approccio, denominato *Neural-Net Approach* prevede la capacità di indurre un comportamento già appreso a seguito di una routine definita da un processo di ricompense e punizioni, collegata in ordine casuale nei suoi vari step. Tale metodologia prevede la sua applicazione in ambito generale, risultando così di ampio utilizzo, ma in grado di fornire risultati e comportamenti non troppo accurati. Il secondo approccio analizzato invece, si basa sulla costruzione di una rete altamente complessa ed organizzata, impostata per imparare una sola e specifica funzione. Questo risulta essere un approccio molto più efficiente, a scapito del fatto che si crea la necessità di programmare e sviluppare la rete per ogni differente applicazione. Con questo studio è stato possibile dimostrare come tramite il ML si potesse arrivare a programmare un computer in grado di giocare a dama in maniera migliore rispetto ad un persona, in seguito ad una fase di training relativamente di breve tempo, in cui vengono forniti pochi paramteri quali le regole del gioco, la direzione, ed una lista ridondante ed incompleta di parametri necessari per il gioco. Questa prima applicazione ha dunque dimostrato le potenzialità dell'apprendimento automatico, verificate in un caso specifico, ma applicabili a molti altri.
Ai giorni d'oggi, il machine learning è entrato a far parte della vita quotidiana delle persone grazie alle innumerevoli applicazioni e l'estensione ai differenti campi possibili. Citandone alcuni vi sono il riconoscimento vocale con conseguente trascrizione del testo, il riconoscimento delle scrittura a mano, la previsione del meteo, l'analisi finanziaria e la guida autonoma di autoveicoli e velivoli. Un altro ambito di grande interesse è quello della classificazione di immagini e video, che vede riscontro anche in applicazioni di tipo medico, nelle diagnosi per immagini, e di tipo ingegneristico.
Soffermandosi nell'ambito ingegneristico, lo strumento utilizzato del machine learning sono le reti neurali, in quanto rappresentano uno strumento dalle ampie potenzialità in

grado di ridurre il carico di lavoro intellettuale degli ingegneri nell'analisi dei dati, oltre che il costo computazionale che sarebbe tipicamente richiesto nello svolgere complesse operazioni di calcolo per la risoluzione di problemi di varia natura. Si tratta quindi di uno strumento non ancora totalmente integrato e consolidato, ma che sicuramente è innovativo e promettente per la produzione di risultati concreti in ambiti in cui gli algoritmi già esistenti possono venire ottimizzati.
Portando l'attenzione sull'ingegneria aerospaziale, è possibile trovare in letteratura una grande quantità di applicazioni del machine learning. In [2] vengono confrontati i casi di impiego riguardanti i veicoli spaziali, così come i razzi ed i satelliti. L'obiettivo è sempre quello di rendere più sicure ed efficienti le varie strutture nel loro complesso utilizzando la grande quantità di dati disponibili grazie al loro funzionamento. Si possono individuare delle aree di interesse pratiche per queste categorie:

- Anomaly Detection
- Collision Avoidance
- Attitude Control
- Satellite Communication
- Default Diagnosis
- Thermoacoustic Instabilities Detection
- Lunar Landing

Nel caso di rilevamento delle anomalie, esistono diversi studi a riguardo basati ciascuno su differenti algoritmi. Lo scopo dell'utilizzo del machine learning consiste nel poter avere un monitoraggio costante ed attivo sulle condizioni del veicolo spaziale, in grado di notare ed identificare eventi e situazioni potenzialmente negativi, così da poter predirre possibili failure dei sistemi e correggere per tempo l'anomalia riscontrata. In [3] viene presentato un tipo di rete neurale ricorrente proprio per adempiere a questo compito, basata su una fase di training offline tramite dati pre-processati, affiancata ad una fase di monitoraggio online. L'unione di queste due fasi ha dimostrato di essere in grado di effettuare una rilevazione ed interpretazione dell'anomalia con accuratezze molto elevate fino ad oltre il 90%.
La prevenzione delle collisioni, riferendosi all'ambiente spaziale, è un altro argomento di grande interesse, soprattutto determinato dal fatto che le orbite attorno al nostro pianeta sono sempre più dense di satelliti ed altri possibili oggetti. In [4] viene sviluppato un sistema di navigazione autonomo basato sul machine learning in grado di gestire una qualsiasi situazione imprevista e di adattarsi di conseguenza. Efficienza ed accuratezza sono stati verificati numericamente all'interno del lavoro, portando come risultato una diminuzione dell'energia consumata del 16.44% nell'evitare la collisione tra due oggetti in prossimità.
Il controllo dell'assetto è un'altra grande problematica per quanto riguarda i veicoli

spaziali a causa della loro elevata massa e di conseguenza inerzia, rendendoli molto difficili da controllare manualmente. Ancor di più il problema si presenta nel caso in cui il sistema dovesse unirsi ad un altro oggetto di massa indefinita mentre orbita nello spazio. Per questi motivi l'implementazione del machine learning risulta essere molto significativa in questo ambito. La sua applicazione, come esposto in [5], avviene con l'utilizzo di un modello di processo decisionale opportuno. Nello studio viene considerata l'indeterminatezza del sistema includendo la velocità angolare, la matrice d'inerzia ignota ed un disturbo periodico, portando a risultati validi in grado di contrapporsi alla matrice d'inerzia.
Per quanto riguarda la comunicazione satellitare, ai giorni nostri questa presenta ancora una forte dipendenza da operazioni manuali, rendendo ancora più cruciale lo sviluppo di modelli di machine learning. In [6] si propone l'utilizzo di una rete neurale artificiale multi-obiettivo per il trasferimento di dati dal satellite alla stazione a terra o ad altri dispositivi. L'algoritmo permette di imparare a riguardo di sè stesso e dell'ambiente intorno, acquisendo così la capacità di adattamento in tempo reale basata sulla simulazione di differenti scenari di missione. Questo si traduce in un'ottimizzazione di trasmissione di dati, riducendo il costo delle operazioni ed aiutando nel gestire la complessità dei sistemi automatici. Le eccellenti prestazioni sono state tramite simulazioni di missioni power saving, dove si ottiene un errore solamente dell'1.77%.
I satelliti artificiali, inoltre risultano essere molto vulnerabili a guasti, soprattutto in quello che è il cuore del sistema, ovvero il sistema di alimentazione. La riparazione e la manutenzione sono operazioni complesse da eseguire, che se non adeguatamente preparate possono portare al fallimento di una operazione in orbita. Proprio in questo il machine learning può portare un contributo essenziale andando a diagnosticare il corretto funzionamento del satellite. L'utilizzo di tre differenti algoritmi di reti neurali artificali viene proposto in [7], in cui l'insieme dei risultati ottenuti viene poi utilizzato per incorporare le prestazioni di ogni modello durante le fasi di training e di prediction. Questo si traduce in una classificazione più accurata dei risultati, dimostrata attraverso set di dati ottenuti da motori aeronautici e da sistemi di alimentazione di satelliti.
Nella propulsione aerospaziale vi è un grande interesse nel poter predirre l'occorrenza di instabilità termoacustiche, per cui attualmente si necessita un costo computazionale molto elevato. L'utilizzo di modelli ed algoritmi di machine learning e reti neurali viene investigato in [8], le quali fasi di training avvengono utilizzando degli specifici modelli fisici. Il risultato è stata la capacità di predirre quali configurazioni fossero instabili, insieme al loro intervallo di frequenza in cui avvengono, in un tempo di calcolo breve e con accuratezze fino al 99.6%.
Il machine learning vede anche utilizzo nel campo dell'allunaggio, il cui impiego può essere multiplo. L'utilizzo di reti neurali artificiali come illustrato in [9], permette al sistema di gestire i retrorazzi nella fase di avvicinamento al suolo per ridurre al minimo il consumo di carburante, ottimizzando così l'operazione con un'accuratezza del modello di quasi il 98%. Inoltre, è stato valutato in [10] l'utilizzo di reti neurali convoluzionali per il rilevamento di ostacoli sulla superificie lunare. Il risultato è stato l'ottenimento di una accuratezza media dell'83%, che sviluppato ulteriormente

può essere implementato in futuro per allunaggi autonomi sulla superficie lunare in sicurezza e con precisione.
Come si può notare, le applicazioni che il machine learning può avere sono numerose e, con lo sviluppo di questa tecnologia, sembrano destinate ad aumentare considerevolmente, visti i grandi vantaggi che questa può apportare.

1.1 Machine learning ed analisi strutturali

Sempre nell'ambito dell'ingegneria aerospaziale, più precisamente nella branchia dell'analisi strutturale l'uso del machine learning sta iniziando ad essere sempre più esteso. Alcuni esempi se ne possono ritrovare in letteratura: in [11] viene sviluppata una rete neurale artificiale per il rilevamento di danneggiamenti e la loro intensità su una struttura; in [12] il machine learning e le reti neurali sono utilizzati per risolvere problemi di natura elastica di strutture di tipo piastra e guscio; in [13] si investiga l'utilizzo ed i valori dei parametri interni della rete neurale necessari per la determinazione dell'accuratezza dei modelli di teoria strutturale.
Un obiettivo comune che si ritrova in queste ricerche ed applicazioni è il voler utilizzare le reti neurali artificiali in sostituzione ai complessi software di calcolo strutturale, poiché in grado di fornire risultati in tempi molto ridotti e con l'utilizzo di risorse hardware ed informatiche meno esigenti.
Infatti, un'analisi strutturale che possa fornire risultati accurati richiede tipicamente l'utilizzo di software commerciali di calcolo dal costo computazionale oneroso, che diventa continuamente più elevato dovuto al sempre più ampio utlizzo di materiali compositi, oltre che alla complessità delle strutture, implicando inoltre un aumento dei tempi necessari di calcolo che può essere notevole a seconda dell'hardware a disposizione per effettuare l'analisi. Il tipo di analisi richiesta, il tipo ed il numero di elementi utilizzati nella modellizzazione della struttura, il modello di teoria utilizzato e il grado della sua espansione, sono tutti elementi che vanno ad influire sulla complessità dei calcoli necessari ad ottenere i risultati desiderati.
Questo aspetto del costo computazionale risulta essere fondamentale e rappresenta una problematica ed una sfida da affrontare molto attuale, in quanto il progresso tecnologico non può avvenire solamente sulla quantità e qualità dei dati ottenuti dalle varie analisi strutturali conducibili su sistemi anche molto complessi, ma deve avvenire rendendo più efficiente anche la metodologia con cui questi risultati vengono ottenuti. L'utilizzo del machine learning permette lo sviluppo di reti neurali in grado di apprendere delle relazioni matematiche tra dati di input e di output, e di applicare queste ultime nel riprodurre risultati accurati, dopo aver fornito un input diverso da quello adoperato in precedenza. Il vantaggio generato è quello che, una volta creata la rete neurale, averne effettuato il training ed averne verificato la validità in termini di affidabilità ed accuratezza, si ottengono i risultati in un tempo e con un costo computazionale irrisori rispetto all'utilizzo di altre metodologie o software, rendendo la metodologia più efficiente.
Il problema che quindi viene affrontato in questa tesi consiste proprio nel creare ed

impostare una rete neurale artificiale tramite linguaggio di programmazione Python che possa fornire risultati di analisi strutturali accurati, sfruttandone i suoi vantaggi in termini di carico computazionale e tempo necessario.

1.2 Struttura dell'elaborato

In questa tesi si intende affrontare la problematica posta in precedenza con un'approccio innovativo andando a sviluppare una rete neurale artificiale apposita ed integrandola nel campo dell'analisi strutturale.
Il primo step consiste nell'introdurre quegli elementi teorici fondamentali per lo svolgimento del lavoro riguardanti i modelli strutturali adottati e le possibili configurazioni ed architettura delle reti neurali artificiali.
Nel campo strutturale, le analisi si basano sulla Carrera Unified Formulation (CUF) per l'analisi ad elementi finiti, la quale grazie alla sua notazione indiciale e gerarchica, permette di ridurre già inizialmente il costo computazionale richiesto grazie all'utilizzo di elementi monodimensionali e bidimensionali per l'ottenimento di soluzioni tipicamente ricavabili tramite teorie tridimensionali e dunque più complesse. Ma questa non è l'unica potenzialità per cui viene utilizzato questo approccio, infatti la CUF riesce ad integrare un metodo denominato assiomatico-asintotico (AAM), il quale si basa sull'utilizzo di ipotesi di natura assiomatica per l'ottenimento di risultati mediante un procedimento asintotico.
Questo particolare metodo viene utilizzato nello svolgimento di questa tesi in due situazioni simili, per l'ottenimento dei Best Theory Diagrams (BTD), ovvero l'individuazione dei modelli di teorie più accurati per la risoluzione strutturale basata su dei parametri di controllo definiti, e per l'ottenimento delle Best Theory Distribution. Quest'ultime riguardano lo studio global/local del sistema in esame e permettono di individuare la miglior distribuzione di modelli di teoria da utilizzare sulla geometria di una data struttura. Questo inoltre rappresenta un punto di forza della formulazione CUF, che introduce la node-dependant kinematics grazie alla quale si rende possibile l'utilizzo di modelli differenti di teorie strutturali per ogni nodo della mesh fem che compone la struttura, potendo così rendere la soluzione più raffinata in punti critici e meno onerosa computazionalmente in nodi meno sottoposti a sollecitazioni.
Entrambi questi metodi legati all'approccio AAM richiedono una capacità di calcolo non indifferente, in quanto devono elaborare combinazioni numerose di teorie e per ognuna di queste valutarne i risultati strutturali.
Viene quindi introdotta nella tesi il machine learning e le reti neurali, andando ad esporre il principio matematico della Backpropagation, di cui ne è fondamenta, i parametri che le compongono, e le differenti architetture che possono essere utilizzate, limitandosi comunque a quelle più rilevanti per lo svolgimento di questo elaborato.
La prima parte pratica presentata è una fase di training, che è stata necessaria per entrare in confidenza con l'utilizzo del codice di calcolo strutturale fornito, basato sulla formulazione CUF. Vengono inoltre riprodotti alcuni risultati consolidati per verificarne l'affidabilità. Insieme a questa, è stata svolta anche una fase di training per

l'utilizzo del linguaggio di programmazione Python, ponendo particolare attenzione alla creazione di reti neurali artificiali ed al loro corretto metodo di training e validazione, andando ad investigare l'influenza dei valori di alcuni parametri fondamentali sui risultati ottenibili.
Nel quinto capitolo, viene esposto come si intende implementare la rete neurale artificiale con le analisi strutturali che verranno eseguite, presentando la metodologia applicata. Successivamente, si presenta la metodologia di sviluppo della rete neurale artificiale che si andrà ad utilizzare, fino all'ottenimento dell'architettura finale e la definizione dei valori per i parametri interni. La peculiarità di questa rete risiede nella capacità di fornire un output multidimensionale, corrispondente in questa applicazione con le prime frequenze naturali del sistema in esame. Entrando nel dettaglio, la rete neurale fornisce gli errori relativi sul calcolo di ognuna delle prime dieci frequenze date da analisi alle vibrazioni libere della struttura, per ogni ordine del modello di teoria strutturale, seguendo l'approccio dettato dal metodo assiomatico-asintotico. L'applicazione viene effettuata su diverse casistiche di struttra di tipo guscio, andando a variarne curvatura, spessore e condizioni di vincolo.
Nel capitolo a seguire, si presenta l'applicazione della rete neurale sviluppata in precedenza, ma utilizzata per la riproduzione di risultati ottenibili tramite analisi strutturali di tipo global/local. In questo caso l'ouput della rete coniste nella distribuzione di teorie strutturali sugli elementi della struttura, con il corrispondente errore relativo sull'accuratezza del modello. Sono presentate differenti casistiche variando i parametri geometrici della struttura di tipo guscio in esame, oltre alle condizioni di vincolo.
Il settimo capitolo studia l'applicazione in un ulteriore tipo di analisi di tipo strutturale, riguardante i failure index in determinati punti della struttura. Questo tipo di implementazione vuole inoltre esplorare ulteriormente le potenzialità della rete neurale, in quanto oltre ai modelli di teorie strutturali come input, vengono fornite anche le coordinate dei punti della struttura su cui si vuole effettuare l'analisi. In questo approccio le dimensioni su cui opera la rete aumentano ulteriormente, andando ad avere una quantità di dati di input per ciascun punto preso in considerazione ed ottenenedo così più output per un numero di punti che può essere superiore all'unità.

Capitolo 2

Modelli strutturali

Nell'ambito dell'analisi strutturale, il metodo ad elementi finiti (FEM) è quello tipicamente più utilizzato, in quanto permette di modellizzare opportunamente una struttura di qualsiasi complessità geometrica e di ottenere risultati più o meno accurati in base al modello di teoria utilizzato.
In [14] viene presentata una metodologia che permette di utilizzare una notazione compatta ed indipendente dalla teoria che si decide di utilizzzare, nella risoluzione strutturale degli elementi di tipo trave (1D) e piastra (2D), che si andranno ad analizzare in questa tesi.

2.1 Principio degli spostamenti virtuali

Prima di procedere con la formulazione CUF, si rende necessario esprimere il principio degli spostamenti virtuali (PVD), essendo questo la base per l'analisi di strutture che possano essere soggette a differenti tipi di carichi e/o vincoli. Tramite questo procedimento, il problema viene affrontato in termini di lavoro, composto da tre contributi principali: il lavoro interno (dovuto al contributo elastico dato dalla deformazione della struttura), il lavoro esterno (dato dai carichi applicati alla struttura) ed il lavoro delle forze d'inerzia (dovuto all'accelerazione a cui è soggetta la massa della struttura). In base alle combinazioni di forze presenti, è possibile individuare tre tipologie differenti di problemi di natura strutturale, che vengono brevemente esposti nelle tre sezioni a seguire.

2.1.1 Analisi statica

La prima tipologia di analisi prevede il contributo delle solo forze elastiche e dei carichi esterni. Il PVD in questo caso prevede il bilanciamento del lavoro esterno con quello esterno secondo la relazione:

$$\delta L_{int} = \delta L_{ext} \tag{2.1}$$

Il lavoro interno e quello esterno possono essere espressi, con l'approssimazione FEM, rispettivamente come:

$$\delta L_{int} = \delta \mathbf{U}^T \mathbf{K} \mathbf{U} \tag{2.2}$$

$$\delta L_{ext} = \delta \mathbf{U}^T \mathbf{P} \tag{2.3}$$

Dove $\mathbf{K}$ rappresenta la matrice di rigidezza, $\mathbf{U}$ il vettore degli spostamenti nodali, $\mathbf{P}$ il vettore dei carichi e $\delta\mathbf{U}$ il vettore degli spostamenti virtuali.
Sostituendo (2.2) e (2.3) in (2.1) e applicando le opportune semplificazioni, si ottiene la classica formulazione dell'equazione di equilibrio:

$$\mathbf{K}\mathbf{U} = \mathbf{P} \tag{2.4}$$

La cui soluzione, non ammettendo movimento rigido delle strutture in modo tale che la matrice di rigidezza sia non singolare ed invertibile, consiste nel ricavare il vettore delle incognite $\mathbf{U}$ tramite l'operazione:

$$\mathbf{U} = \mathbf{K}^{-1}\mathbf{P} \tag{2.5}$$

2.1.2 Analisi vibrazioni libere

L'analisi di vibrazioni libere è il primo caso di analisi dinamica, in cui non è presente una forzante esterna, e viene considerato l'equilibrio tra le forze elastiche interne e quelle di inerzia. Il PVD in questo caso viene espresso come:

$$\delta L_{int} = \delta L_{ine} \tag{2.6}$$

La variazione virtuale delle forze d'inerzia nell'approssimazione FEM viene espressa come:

$$\delta L_{ine} = \delta \mathbf{U}^T \mathbf{M} \ddot{\mathbf{U}} \tag{2.7}$$

Richiamando (2.2) e sostituendola insieme a (2.7) in (2.6) si ottiene l'equazione di equilibrio:

$$\delta \mathbf{U}^T \mathbf{K} \mathbf{U} = -\delta \mathbf{U}^T \mathbf{M} \ddot{\mathbf{U}} \tag{2.8}$$

$$\mathbf{M}\ddot{\mathbf{U}} + \mathbf{K}\mathbf{U} = 0 \tag{2.9}$$

Dove $\mathbf{M}$ è la matrice delle masse e le soluzioni $\mathbf{U}$ sono ricavabili tramite un problema agli autovalori, trattandosi di un sistema omogeneo.
Un metodo semplice per la sua risoluzione si ottiene considerando la forma armonica:

$$\mathbf{U} = \bar{\mathbf{U}} e^{i\omega t} \tag{2.10}$$

2.1.3 Analisi dinamica

L'ultimo caso consiste in un'analisi dinamica completa e prende in considerazione tutte e tre le tipologie di lavoro espresse in precedenza. L'equazione del PVD viene dunque scritta nella forma:

$$\delta L_{ine} + \delta L_{int} = \delta L_{ext} \tag{2.11}$$

Sostituendo al suo interno le relazioni precedenti (2.7), (2.2) e (2.3), si ottiene l'equazione di equilibrio:

$$\delta \mathbf{U}^T \mathbf{M} \ddot{\mathbf{U}} + \delta \mathbf{U}^T \mathbf{K} \mathbf{U} = \delta \mathbf{U}^T \mathbf{P} \tag{2.12}$$

$$\mathbf{M} \ddot{\mathbf{U}} + \mathbf{K} \mathbf{U} = \mathbf{P} \tag{2.13}$$

Le cui soluzioni $\mathbf{U}$ sono espresse nel dominio del tempo e possono essere valutate con differenti approcci, non affrontati all'interno di questa tesi.

2.2 Carrera Unified Formulation

La CUF, come anticipato, utilizza una notazione compatta nell'esprimere i campi di spostamento lungo la sezione trasversale, per il caso della trave, e lungo lo spessore, per il caso della piastra, utilizzando funzioni matematiche di forma ed ordine arbitrari. Questo avviene sfruttando i *fundamental nucleus* (FN) dei vari vettori e matrici FEM. Questi nuclei vengono derivati direttamente dalle equazioni elastiche 3D, tramite il principio degli spostamenti virtuali, con la possibilità di ottenerli direttamente per il caso dimensionale di interesse. Un grosso vantaggio di questa formulazione è il fatto che gli elementi 1D e 2D così ricavati, sono in grado di fornire risultati tipicamente ottenibili tramite teorie 3D, ma con un costo computazionale decisamente inferiore.
Questi nuclei fondamentali corrispondono a delle matrici 3x3, la cui forma non cambia nel caso di problemi 1D, 2D o 3D. Ogni FN è espresso con l'utilizzo di quattro indici ($\tau,\ s,\ i,\ j$), con i primi due derivanti dalle teorie strutturali e gli ultimi due derivanti dalle procedure FEM.

2.2.1 Elemento Barra

Si vuole ora descrivere la procedura per ricavare la matrice di rigidezza di un elemento barra, rappresentativo del problema unidimensionale, ed il corrispettivo nucleo fondamentale.
Vengono introdotte le relazioni per spostamento e rispettiva variazione virtuale, utilizzando l'indice i per il primo e j per il secondo.

$$u_y(y) = N_i(y) u_{y_i} \tag{2.14}$$

$$\delta u_y(y) = N_j(y)\delta u_{y_j} \tag{2.15}$$

Dove le u rappresentano gli spostamenti e le N le funzioni di forma. Quest'ultime dipendono dalla geometria dell'elemento e dal numero di nodi NE che lo compongono. Inoltre, gli indici ripetuti denotano una sommatoria, pertanto si possono riscrivere le due relazioni nella forma seguente:

$$u_y(y) = N_i(y)u_{y_i} = \sum_{i=1}^{N_{NE}} N_i(y)u_{y_i} = N_1u_{y_1} + N_2u_{y_2} + ... + N_{N_{NE}}u_{y_{N_{NE}}} \tag{2.16}$$

$$\delta u_y(y) = N_j(y)\delta u_{y_j} = \sum_{j=1}^{N_{NE}} N_j(y)\delta u_{y_j} = N_1\delta u_{y_1} + N_2\delta u_{y_2} + ... + N_{N_{NE}}\delta u_{y_{N_{NE}}} \tag{2.17}$$

Analogamente, si possono scrivere le relazioni per deformazione e rispettiva variazione virtuale:

$$\epsilon = N_{i,y}u_{y_i} \tag{2.18}$$

$$\delta\epsilon = N_{j,y}\delta u_{y_j} \tag{2.19}$$

Dove $N_{i,y} = dN/dy$.
A questo punto, ricordando anche la relazione $\sigma = \epsilon E$, è possibile scrivere la variazione virtuale del lavoro interno come:

$$\begin{aligned} \delta L_{int} &= \int_V \delta\epsilon^T\sigma \; dV \\ &= \int_V \delta\epsilon^T E\epsilon \; dV \\ &= \delta u_{y_j}\Big(\int_V N_{j,y}EN_{i,y} \; dV\Big)u_{y_i} \\ &= \delta u_{y_j} \; k^{ij} \; u_{y_i} \end{aligned} \tag{2.20}$$

Si definisce quindi il nucleo fondamentale dell'elemento unidimensionale barra come:

$$k^{ij} = \int_V N_{j,y} \; E \; N_{i,y} \; dV \tag{2.21}$$

Si nota come questo sia invariante rispetto al numero dei nodi che compongono l'elemento e rispetto la scelta delle funzioni di forma.
Un esempio applicativo molto semplice è riportato in [14], in cui si fa riferimento ad una barra, riportata in figura 2.1, costituita dai soli due nodi all'estremità.

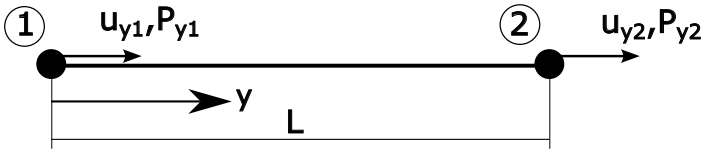


Figura 2.1: Elemento barra a 2 nodi.

Essendo l'elemento composto da due nodi, si assumono delle funzioni di forma lineari e si possono valutare i nuclei fondamentali:

$$k^{11} = \int_V N_{1,y}\ E\ N_{1,y}\ dV = \frac{1}{L} E \frac{1}{L} AL = EA/L \tag{2.22}$$

$$k^{12} = \int_V N_{2,y}\ E\ N_{1,y}\ dV = -\frac{1}{L} E \frac{1}{L} AL = -EA/L \tag{2.23}$$

$$k^{21} = \int_V N_{1,y}\ E\ N_{2,y}\ dV = -\frac{1}{L} E \frac{1}{L} AL = -EA/L \tag{2.24}$$

$$k^{22} = \int_V N_{2,y}\ E\ N_{2,y}\ dV = \frac{1}{L} E \frac{1}{L} AL = EA/L \tag{2.25}$$

L'ultimo passaggio consiste nell'assemblare la matrice di rigidezza globale posizionando i vari nuclei fondamentali nella corretta posizione fornita dagli indici i e j.

$$\mathbf{K} = \begin{bmatrix} k^{11} & k^{12} \\ k^{21} & k^{22} \end{bmatrix} = \frac{EA}{L} \begin{bmatrix} 1 & -1 \\ -1 & 1 \end{bmatrix} \tag{2.26}$$

Il risultato così ottenuto coincide con quello derivabile dalle teorie classiche ed è facilmente estendibile ad elementi con una moltitudine di nodi maggiore.
Il campo di spostamento considerato finora è unidimensionale, considerato costante lungo la sezione trasversale. Per una descrizione più accurata, si utilizza un campo di spostamenti tridimensionale, esposto in (2.27), andando a considerare un campo non costante lungo la sezione, definito da $u_{y_i}(x,z)$ ed introducendo una generica funzione di espansione $F_\tau(x,z)$.

$$u_y(x,y,z) = N_i(y) u_{y_i}(x,z) \tag{2.27}$$

Si può dunque scrivere lo spostamento lungo la direzione trasversale come:

$$\begin{aligned} u_{y_i}(x,z) &= F_\tau(x,z) u_{y_{\tau i}} \\ &= F_1(x,z) u_{y_{1i}} + F_2(x,z) u_{y_{2i}} + ... + F_\tau(x,z) u_{y_{\tau i}} + ... + F_m(x,z) u_{y_{Mi}} \end{aligned} \tag{2.28}$$

Dove M rappresenta il numero di termini dell'espansione e viene scelto arbitrariamente. Si possono andare a riscrivere le equazioni (2.14) e (2.15) con l'utilizzo dell'indice s al posto di τ all'interno della variazione virtuale.

$$u_y(x,y,z) = N_i(y)F_\tau(x,z)u_{y_{\tau i}} \tag{2.29}$$

$$\delta u_y(x,y,z) = N_j(y)F_s(x,z)\delta u_{y_{sj}} \tag{2.30}$$

Tramite questa equazione di campo di spostamento 3D si nota come venga considerata un'espansione delle soluzioni sulla sezione trasversale che differisce dall'utilizzo delle funzioni di forma per espandere le soluzioni dei nodi lungo l'asse.
Si possono riscrivere anche le relazioni per deformazioni e tensioni, utilizzando la matrice **b** contentente gli operatori differenziali e la matrice **C** contenente le rigidezze del materiale, risultando:

$$\epsilon(x,y,z) = \mathbf{b}N_i(y)F_\tau(x,z)u_{y_{\tau i}} \tag{2.31}$$

$$\sigma(x,y,z) = \mathbf{C}\mathbf{b}N_i(y)F_\tau(x,z)u_{y_{\tau i}} \tag{2.32}$$

Sostituendo gli indici analogamente a quanto fatto in (2.30), si scrive anche la variazione virtuale delle deformazioni:

$$\delta\epsilon(x,y,z) = \mathbf{b}N_j(y)F_s(x,z)\delta u_{y_{sj}} \tag{2.33}$$

A questo punto è possibile scrivere il nucleo fondamentale per il caso in cui il campo di spostamenti sia tridimensionale, seguendo lo stesso procedimento utilizzato in precedenza.
Quello che si ottiene è una matrice 3 x 3 individuata da:

$$\mathbf{k}^{\tau sij} = \int_V F_s(x,z)N_j(y)\mathbf{b}^T\mathbf{C}\mathbf{b}N_i(y)F_\tau(x,z)\ dV \tag{2.34}$$

2.2.2 Tecnica di assemblaggio CUF

L'utilizzo della CUF permette di assemblare le matrici facilmente implementando dei codici di calcolo funzionanti tramite quattro cicli imposti sugli indici i, j, τ, s. Il procedimento necessario, di cui viene riportato uno schema in figura 2.2, consiste nell'agire inizialmente sugli indici τ e s per la costruzione della matrice di ogni singolo nodo identificato da i e j, per ogni termine M dell'espansione. Successivamente, si agisce sugli indici i e j per la costruzione della matrice degli elementi, la quale viene sfruttata nell'ultimo passaggio per la creazione della matrice globale della struttura.

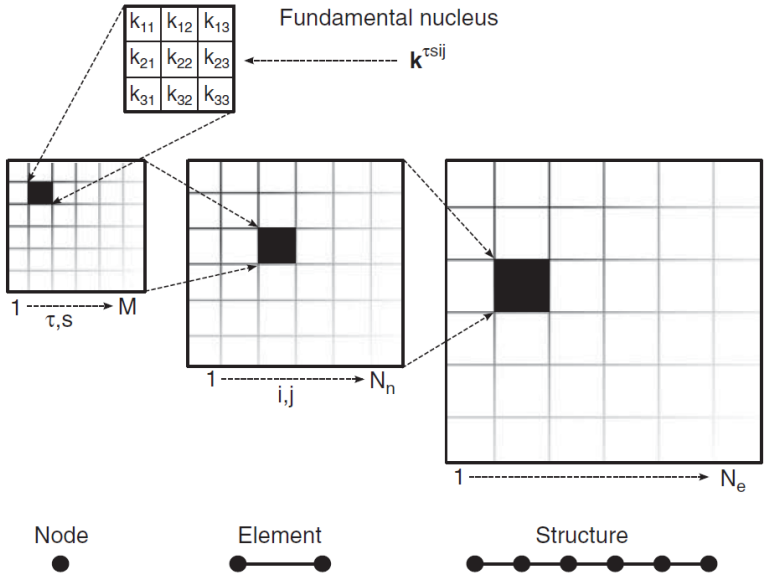


Figura 2.2: Schema assemblaggio CUF [14].

2.2.3 Modelli 1D, 2D e 3D

Quanto esposto per i modelli unidimensionali, può essere facilmente esteso anche ad altre teorie strutturali bidimensionali e tridimensionali, grazie alla notazione indiciale introdotta con la CUF. Quello che cambia sono il tipo di funzioni di forma utilizzate:

$$\begin{aligned} &1D \rightarrow N_i(y) \\ &2D \rightarrow N_i(x,y) \\ &3D \rightarrow N_i(x,y,z) \end{aligned} \tag{2.35}$$

Insieme alle funzioni di espansione necessarie:

$$\begin{aligned} &1D \rightarrow F_\tau(x,z) \\ &2D \rightarrow F_\tau(z) \\ &3D \rightarrow 1 \end{aligned} \tag{2.36}$$

Si può dunque esprimere il campo di spostamento, che sarà sempre di tipo tridimensionale, per ogni tipo di modello FEM:

$$\begin{aligned} &1D \rightarrow \mathbf{u}(x,y,z) = N_i(y)F_\tau(x,z)\mathbf{u}_{\tau i} \\ &2D \rightarrow \mathbf{u}(x,y,z) = N_i(x,y)F_\tau(z)\mathbf{u}_{\tau i} \\ &3D \rightarrow \mathbf{u}(x,y,z) = N_i(x,y,z)\mathbf{u}_i \end{aligned} \tag{2.37}$$

In questa sezione sono state presentate varie relazioni che prevedono l'utilizzo di funzioni di espansione e funzioni di forma al loro interno. Le prime saranno affrontate nella sezione a seguire di questa tesi, mentre per le ultime possono essere trovati maggiori dettagli in [15].

2.2.4 Modello trave

Il modello trave è definito da una struttura tridimensionale in cui una delle sue dimensioni, solitamente individuata dalla lunghezza L, risulta essere di almeno un ordine di grandezza maggiore rispetto alle rimanenti due ed in cui è valida la relazione $L/d \gg 1$, in cui d rappresenta la lunghezza caratteristica trasverale. In questa maniera è possibile ridurre il problema ad un modello 1D con riferimento all'asse della trave. In figura 2.3 viene presentata la geometria del modello comprensivo delle dimensioni b, h, L ed il sistema di riferimento adottato.

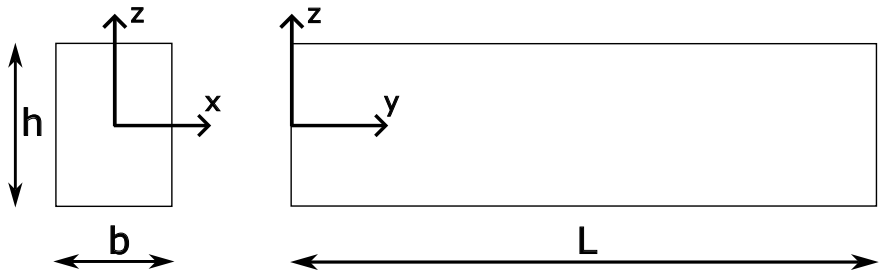


Figura 2.3: Modello trave.

Il vettore degli spostamenti $\mathbf{u}$ in campo 3D viene espresso con l'utilizzo delle tre componenti lungo gli assi:

$$\mathbf{u}(x, y, z) = \{u_x \;\; u_y \;\; u_z\}^T \tag{2.38}$$

Il modello trave più semplice è quello della *Euler-Bernoulli Beam Theory* (EBBT), basato su tre ipotesi fondamentali che affermano che la sezione trasversale rimane rigida sul suo piano, oltre a rimanere rettilinea e perpendicolare all'asse della trave a deformazione avvenuta.
Il campo di spostamenti ottenuto dal modello EBBT, in cui il pedice 0 rappresenta lo spostamento relativo all'asse della trave, è quanto segue:

$$\begin{aligned} u_x &= u_{x_0} \\ u_y &= u_{y_0} - \frac{\partial u_{x_0}}{\partial y} x - \frac{\partial u_{z_0}}{\partial y} z \\ u_z &= u_{z_0} \end{aligned} \tag{2.39}$$

Una diretta evoluzione di questa teoria è la *Timoshenko Beam Theory* (TBT), nella quale viene rimossa una delle ipotesi formulate da Eulero, ammettendo che la sezione trasversale possa non essere più perpendicolare all'asse della trave a deformazione avvenuta. In questa maniera si considerano anche gli effetti degli scorrimenti a taglio γ_{xy} e γ_{yz} che non possono più essere trascurati.
Il campo di spostamenti ottenuto dal modello TBT, comprensivo delle rotazioni ϕ

attorno agli assi y, z, viene espresso dalla seguenti relazioni:

$$\begin{aligned} u_x(x,y,z) &= u_{x_0}(y) \\ u_y(x,y,z) &= u_{y_0}(y) + \phi_z(y)x + \phi_x(y)z \\ u_z(x,y,z) &= u_{z_0}(y) \end{aligned} \tag{2.40}$$

Ulteriori evoluzioni di queste teorie avvengono con l'arricchimento del campo di spostamenti, come accade nell'ambito CUF in cui, utilizzando delle funzioni di espansione di ordine elevato, si può ottenere un risultato molto simile alla soluzione esatta 3D. La scelta delle funzioni di espansione utilizzate in questo lavoro di tesi ricade sui polinomi di Taylor e di Lagrange, che vengono esposti a seguire.
La formulazione della teoria avanzata tramite notazione CUF viene dunque scritta come:

$$\mathbf{u}(x,y,z) = N_i(y)F_\tau(x,z)\mathbf{u}_{\tau i} \tag{2.41}$$

Analogamente a quanto fatto per gli spostamenti, si possono descrivere i campi di tensioni $\boldsymbol{\sigma}$ e deformazioni $\boldsymbol{\epsilon}$ utilizzando le loro componenti:

$$\begin{aligned} \boldsymbol{\sigma} &= \{\sigma_{xx} \quad \sigma_{yy} \quad \sigma_{zz} \quad \sigma_{xz} \quad \sigma_{yz} \quad \sigma_{xy}\}^T \\ \boldsymbol{\epsilon} &= \{\epsilon_{xx} \quad \epsilon_{yy} \quad \epsilon_{zz} \quad \epsilon_{xz} \quad \epsilon_{yz} \quad \epsilon_{xy}\}^T \end{aligned} \tag{2.42}$$

Viene introdotta la matrice degli operatori differenziali $\mathbf{b}$:

$$\mathbf{b}_{trave} = \begin{bmatrix} \partial_x & 0 & 0 \\ 0 & \partial_y & 0 \\ 0 & 0 & \partial_z \\ \partial_z & 0 & \partial_x \\ 0 & \partial_z & \partial_y \\ \partial_y & \partial_x & 0 \end{bmatrix} \tag{2.43}$$

In cui il simbolo ∂_x indica l'operazione di differenziazione $\partial_x = \frac{\partial(.)}{\partial x}$, analogo per tutte le componenti.
Si può dunque scrivere la relazione geometrica:

$$\boldsymbol{\epsilon} = \mathbf{bu} \tag{2.44}$$

Ed, in maniera analoga, introducendo la matrice di elasticità $\mathbf{C}$, la quale varia in base al materiale in utilizzo e di cui se ne può trovare forma esplicita in [14], si definisce la relazione costitutiva, o legge di Hooke, come:

$$\boldsymbol{\sigma} = \mathbf{C}\boldsymbol{\epsilon} \tag{2.45}$$

2.2.5 Modello piastra e guscio

Il modello piastra è definito da una struttura tridimensionale in cui una delle sue tre dimensioni, solitamente individuata dallo spessore h, risulta essere inferiore di almeno un ordine di grandezza rispetto alle altre due. In questa maniera, è possibile ridurre il problema ad un modello 2D con riferimento alla superficie media della piastra. In figura 2.4 viene rappresentata la geometria del modello con il rispettivo sistema di riferimento e le sue tre dimensioni a, b, h.

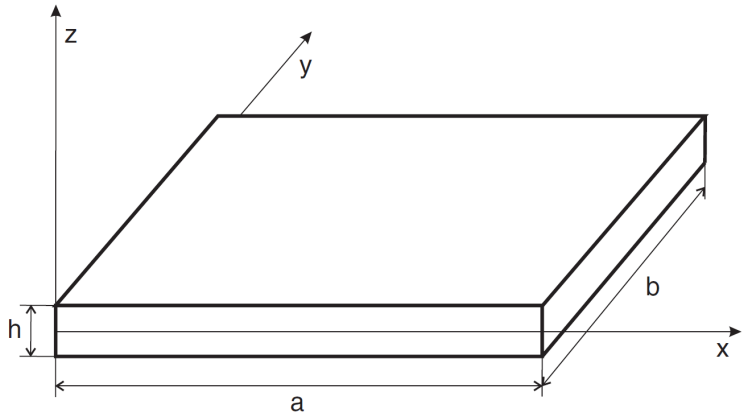


Figura 2.4: Modello Piastra [14].

Il vettore degli spostamenti **u** del caso piastra in campo 3D è espresso, con l'utilizzo delle tre componenti lungo gli assi, come segue:

$$\mathbf{u}(x, y, z) = \{u_x \quad u_y \quad u_z\}^T \tag{2.46}$$

Il modello della piastra di Kirchoff-Love, denominato *Classical Plate Theory* (CPT), si basa sulle tre ipotesi fondamentali che prevedono che i segmenti perpendicolari alla superficie media della piastra rimangano tali anche a deformazione avvenuta e che lo spessore della piastra rimanga della stessa lunghezza e ruoti in modo da tale da rimanere perpendicolare alla superficie media a seguito della deformazione.
Il campo di spostamenti ottenuto dal modello CPT, in cui il pedice 0 rappresenta lo spostamento relativo alla superficie media, è quanto segue:

$$\begin{aligned} u_x &= u_{x_0} - \frac{\partial u_{z_0}}{\partial x} z \\ u_y &= u_{y_0} - \frac{\partial u_{z_0}}{\partial y} z \\ u_z &= u_{z_0} \end{aligned} \tag{2.47}$$

Una diretta evoluzione di questa teoria è formulata da Reissner-Mindlin, chiamata *First-order Shear Deformation Theory* (FSDT), nella quale viene rimossa la terza ipotesi di Kirchoff, pertanto lo spessore della piastra non rimane più perpendicolare alla

superficie di riferimento a deformazione avvenuta. In questa maniera si considerano anche gli effetti di primo ordine del taglio, includendo nella teoria gli scorrimenti γ_{xz} e γ_{yz}.
Il campo di spostamenti del modello FSDT include le rotazione ϕ ed è il seguente:

$$\begin{aligned} u_x(x,y,z) &= u_{x_0}(x,y) + \phi_x(x,y)z \\ u_y(x,y,z) &= u_{y_0}(x,y) + \phi_y(x,y)z \\ u_z(x,y,z) &= u_{z_0}(x,y) \end{aligned} \tag{2.48}$$

L'evoluzione successiva nelle teorie avanzate, tra cui quella utilizzata nell'ambito CUF, prevede l'arricchimento del campo di spostamenti, che avviene mediante espansioni delle teorie 1D o 2D. Questo procedimento, come illustrato in [16], permette di ottenere la soluzione esatta 3D utilizzando un'espansione infinita. Non essendo chiaramente possibile l'utilizzo di un'espansione infinita, l'approccio consiste nel troncare ad un ordine elevato l'espansione così da ottenere risultati con un'ottima accuratezza. Anche in questo caso ci sono diverse funzioni di espansione che possono essere utilizzate, ma nello svolgimento di questa tesi, in particolare nella risoluzione di modelli piastra tramite CUF, vengono considerate solo espansioni di Taylor, che vengono esposte nella sezione a seguire.
Si può dunque scrivere la formulazione CUF del modello piastra come:

$$\mathbf{u}(x,y,z) = N_i(x,y)F_\tau(z)\mathbf{u}_{\tau i} \tag{2.49}$$

Analogamente a quanto fatto per gli spostamenti, e con la stessa metodologia utilizzata per il modello trave, si esprimono tensioni $\boldsymbol{\sigma}$ e deformazioni $\boldsymbol{\epsilon}$ in forma vettoriale:

$$\begin{aligned} \boldsymbol{\sigma} &= \{\sigma_{xx} \quad \sigma_{yy} \quad \sigma_{zz} \quad \sigma_{xz} \quad \sigma_{yz} \quad \sigma_{xy}\}^T \\ \boldsymbol{\epsilon} &= \{\epsilon_{xx} \quad \epsilon_{yy} \quad \epsilon_{zz} \quad \epsilon_{xz} \quad \epsilon_{yz} \quad \epsilon_{xy}\}^T \end{aligned} \tag{2.50}$$

Viene introdotta la matrice degli operatori differenziali $\mathbf{b}$:

$$\mathbf{b}_{piastra} = \begin{bmatrix} \partial_x & 0 & 0 \\ 0 & \partial_y & 0 \\ 0 & 0 & \partial_z \\ \partial_z & 0 & \partial_x \\ 0 & \partial_z & \partial_y \\ \partial_y & \partial_x & 0 \end{bmatrix} \tag{2.51}$$

Dove il simbolo ∂_x indica l'operazione $\partial_x = \frac{\partial(.)}{\partial x}$, ed analogamente per le altre componenti.
Grazie a questa matrice è possibile definire la relazione geometrica:

$$\boldsymbol{\epsilon} = \mathbf{bu} \tag{2.52}$$

Introducendo poi la matrice di elasticità $\mathbf{C}$, definita dal tipo di materiale preso in considerazione e di cui se ne può trovare la forma esplicita in diverse fonti in letteratura come ad esempio in [17], si definisce la relazione costitutiva:

$$\boldsymbol{\sigma} = \mathbf{C}\boldsymbol{\epsilon} \tag{2.53}$$

Lo stesso tipo di teoria viene utilizzato all'interno di questa tesi nel caso di struttura a guscio, trattandosi sempre di un modello 2D. Le differenze alla base consistono nel fatto che vengano utilizzate delle coordinate curvilinee α e β al posto di (x, y), oltre alla presenza di curvature più o meno accentuate R_α e R_β. Una rappresentazione del modello guscio viene riportata in figura 2.5.

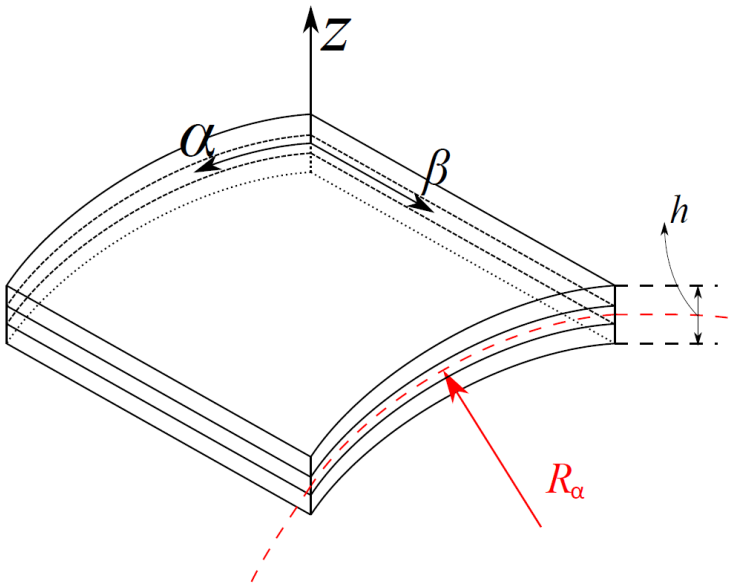


Figura 2.5: Modello Guscio [18].

Soffermandosi sulla sua formulazione CUF, questa diventa quindi del tipo:

$$\mathbf{u}(\alpha, \beta, z) = N_i(\alpha, \beta)F_\tau(z)\mathbf{u}_{\tau i} \tag{2.54}$$

Le formulazioni vettoriali per il campo di spostamenti, tensioni e deformazioni subiscono altresì la modifica delle coordinate x e y con le corrispettive coordinate curvilinee α e β. L'effetto delle curvature si manifesta nella derivazione delle deformazioni a partire dagli spostamenti e, conseguentemente nella derivazione delle tensioni, dovuto all'introduzione dei parametri $H_\alpha = (1 + \frac{z}{R_\alpha})$ e $H_\beta = (1 + \frac{z}{R_\beta})$, che vanno così a

modificare la matrice degli operatori differenziali **b**:

$$\mathbf{b}_{guscio} = \begin{bmatrix} \frac{\partial_\alpha}{H_\alpha} & 0 & \frac{1}{H_\alpha R_\alpha} \\ 0 & \frac{\partial_\beta}{H_\beta} & \frac{1}{H_\beta R_\beta} \\ 0 & 0 & \partial_z \\ \partial_z - \frac{1}{H_\alpha R_\alpha} & 0 & \frac{\partial_\alpha}{H_\alpha} \\ 0 & \partial_z - \frac{1}{H_\beta R_\beta} & \frac{\partial_\beta}{H_\beta} \\ \frac{\partial_\beta}{H_\beta} & \frac{\partial_\alpha}{H_\alpha} & 0 \end{bmatrix} \tag{2.55}$$

2.3 Funzioni di espansione

All'interno della formulazione CUF possono essere implementate differenti funzioni di espansione. Affrontando il problema con un approccio basato solamente sugli spostamenti, possono essere utilizzate funzioni polinomiali, armoniche ed esponenziali per descrivere il campo lungo la sezione trasversale della struttura. Le funzioni più facilmente implementabili e più comunemente utilizzate nell'ambito CUF, sono quelle polinomiali quali l'espansione di Lagrange (LE) e l'espansione di Taylor (TE), che vengono ora riportate.

2.3.1 Espansioni di Taylor

Le funzioni di espansione di Taylor (TE) introducono un sistema gerarchico nell'interpolazione della sezione trasversale, dove all'aumentare dell'ordine di espansione, si aggiungono specifici termini all'espansione. I polinomi considerati per modelli unidimensionali sono del tipo $F_\tau(x,z) = x^i z^j$, dove i e j sono degli interi positivi con valore crescente fino a quello del grado di espansione N. Queste funzioni vengono rappresentate nella tabella 2.1 insieme all'ordine N a cui si riferiscono e al numero di termini M che le compongono.

Tabella 2.1: Polinomi espansione di Taylor modello 1D.

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	$F_7 = x^3\ F_8 = x^2 z\ F_9 = xz^2\ F_{10} = z^3$
⋮	⋮	⋮
N	$(N+1)(N+2)/2$	$F_{(N^2+N+2)/2} = x^N\ \ldots\ F_{(N+1)(N+2)/2} = z^N$

Un generico campo di spostamento, ad esempio con ordine $N = 2$ come presentato nella (2.56), presenta quindi diciotto spostamenti come variabili ed è composto da

tre termini costanti ($N = 0$), sei termini lineari ($N = 1$) e nove termini parabolici ($N = 2$).

$$\begin{aligned} u_x &= u_{x_1} + xu_{x_2} + zu_{x_3} + x^2u_{x_4} + xzu_{x_5} + z^2u_{x_6} \\ u_y &= u_{y_1} + xu_{y_2} + zu_{y_3} + x^2u_{y_4} + xzu_{y_5} + z^2u_{y_6} \\ u_z &= \underbrace{u_{z_1}}_{\text{N=0}} + \underbrace{xu_{z_2} + zu_{z_3}}_{\text{N=1}} + \underbrace{x^2u_{z_4} + xzu_{z_5} + z^2u_{z_6}}_{\text{N=2}} \end{aligned} \tag{2.56}$$

Si può notare come le teorie classiche della trave, quali Eulero e Timoshenko, siano riconducibili a dei casi particolari dell'espansione di ordine $N = 1$.
Nel caso bidimensionale, la funzione di espansione dipende dalla sola coordinata z, ed è del tipo $F_\tau(z) = z^i$. In questo caso, il pedice τ assume valori compresi tra 0 e N, corrispondente all'ordine del polinomio che viene utilizzato lungo la coordinata. Una rappresentazione di queste funzioni è riportata in tabella 2.2, in cui analogamente N indica l'ordine di espansione e M il numero di termini.

Tabella 2.2: Polinomi espansione di Taylor modello 2D.

N	M	F_τ
0	1	$F_0 = 1$
1	2	$F_1 = z$
2	3	$F_2 = z^2$
3	4	$F_3 = z^3$
$\vdots$	$\vdots$	$\vdots$
N	N+1	$F_N = z^N$

Analogamente, il campo di spostamenti ottenuto con l'utilizzo di queste funzioni di spostamento è del tipo:

$$\begin{aligned} u_x &= u_{x_0} + zu_{x_1} + z^2u_{x_2} + z^3u_{x_3} + ... + z^Nu_{x_N} \\ u_y &= u_{y_0} + zu_{y_1} + z^2u_{y_2} + z^3u_{y_3} + ... + z^Nu_{y_N} \\ u_z &= \underbrace{u_{z_0}}_{\text{N=0}} + \underbrace{zu_{z_1}}_{\text{N=1}} + \underbrace{z^2u_{z_2}}_{\text{N=2}} + \underbrace{z^3u_{z_3}}_{\text{N=3}} + ... + \underbrace{z^Nu_{z_N}}_{\text{N}} \end{aligned} \tag{2.57}$$

2.3.2 Espansioni di Lagrange

I modelli 1D di espansioni di Lagrange, indicati come modelli LE, sono caratterizzati da tre principali proprietà:

1. Le variabili del modello LE e le condizioni al contorno possono essere localizzate sulla superificie fisica della struttura. Questo si traduce in una particolare utilità nell'accoppiamento CAD-FEM.

2. Le variabili incognite del problema sono solamente componenti di spostamento. Non sono presenti rotazioni o altre variabili di ordine maggiore nel descrivere il campo di spostamento del modello LE.

3. Altri modelli più raffinati localmente possono essere implementati con facilità, dato che la distribuzione dei polinomi di Lagrange può essere impostata arbitrariamente sulla sezione trasversale.

Grazie a queste proprietà, i modelli 1D CUF LE presentano elevate capacità ed accuratezza confrontabili con i modelli 3D, ad un costo computazionale decisamente inferiore.
Nel descrivere la sezione trasversale, vengono impiegati polinomi di Lagrange 2D, definiti tramite delle geometrie normalizzate e denominati in base al numero di nodi utilizzati. Diverse geometrie sono riportate in [14], ma in questa trattazione vengono riportate solo la geometria quadrangolare a quattro e nove nodi, essendo quella che verrà poi utilizzata.
L'elemento quadrangolare a quattro nodi (L4) viene rappresentato in figura 2.6 e la sua normalizzazione avviene con l'utilizzo del polinomio di Lagrange seguente:

$$F_\tau = \frac{1}{4}(1+\alpha\alpha_\tau)(1+\beta\beta_\tau), \qquad \tau = 1,2,3,4 \tag{2.58}$$

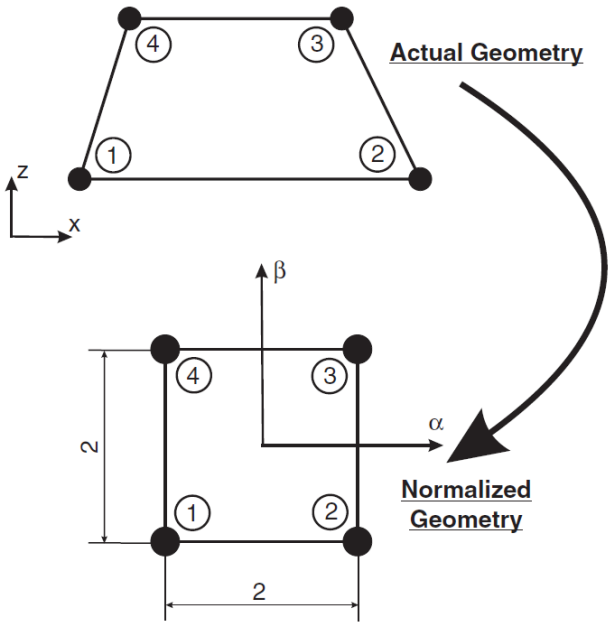


Figura 2.6: Normalizzazione elemento quadrangolare a quattro nodi (L4) tramite polinomi di Lagrange [14].

L'elemento L4 può essere visto quindi come la somma tra termini di espansione lineare $(1, \alpha, \beta)$ e un termine bilineare $(\alpha\beta)$. Le sue coordinate normalizzate dei nodi sono riportate in tabella 2.3.

Tabella 2.3: Coordinate normalizzate dei nodi elemento L4.

Nodo	α_τ	β_τ
1	-1	-1
2	1	-1
3	1	1
4	-1	1

L'elemento quadrangolare a nove nodi (L9), rappresentato in figura 2.7, viene invece ottenuto tramite interpolazioni con tre differenti polinomi di Lagrange che sono:

$$\begin{aligned} F_\tau &= \frac{1}{4}(\alpha^2 + \alpha\alpha_\tau)(\beta^2 + \beta\beta_\tau), && \tau = 1,3,5,7 \\ F_\tau &= \frac{1}{2}\beta_\tau^2(\beta^2 - \beta\beta_\tau)(1-\alpha^2) + \frac{1}{2}\alpha_\tau^2(\alpha^2 - \alpha\alpha_\tau)(1-\beta^2), && \tau = 2,4,6,8 \\ F_\tau &= (1-\alpha^2)(1-\beta^2), && \tau = 9 \end{aligned} \tag{2.59}$$

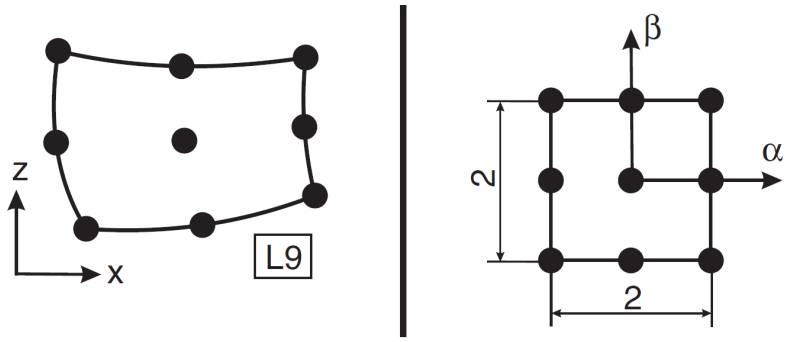


Figura 2.7: Normalizzazione elemento quadrangolare a nove nodi (L9) tramite polinomi di Lagrange [14].

L'elemento L9 può quindi essere visto come una somma di espansione parabolica con due termini cubici $(\alpha\beta^2, \alpha^2\beta)$ ed un termine di quarto ordine $(\alpha^2\beta^2)$. Le coordinate nodali dell'elemento normalizzato sono riportate in tabella 2.4.

Il campo di spostamento viene dunque definito tenendo conto dello spostamento di ogni nodo. Ad esempio, nel caso di elemento L4 si ottiene:

$$\begin{aligned} u_x &= F_1 u_{x1} + F_2 u_{x2} + F_3 u_{x3} + F_4 u_{x4} \\ u_y &= F_1 u_{y1} + F_2 u_{y2} + F_3 u_{y3} + F_4 u_{y4} \\ u_z &= F_1 u_{z1} + F_2 u_{z2} + F_3 u_{z3} + F_4 u_{z4} \end{aligned} \tag{2.60}$$

Tabella 2.4: Coordinate normalizzate dei nodi elemento L9.

Nodo	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

Dove le varie u rappresentano gli spostamenti del nodo lungo la direzione indicata.

2.4 Global/Local

Un importante punto di forza della formulazione CUF è la possibilità di svolgere analisi di tipo global/local con l'utilizzo della Node-Dependent Kinematics (NDK).
Il campo di spostamenti tridimensionale, come esposto in precedenza, è il risultato di una combinazione tra una funzione di forma ed una funzione di espansione. Prendendo in considerazione il caso di formulazione CUF 2D, la funzione di espansione è definita lungo lo spessore della struttura come $F_\tau(z)$ e questa può essere definita diversamente per ciascuno dei nodi che compongono la mesh del modello FEM. Questo permette di assegnare differenti ordini e funzioni di espansione e, conseguentemente, diverse teorie strutturali da utilizzare per la risoluzione, con l'utilizzo di un unico modello numerico. La struttura in esame quindi, viene definita seguendo questa metodologia attraverso dei sotto-domini, in cui ognuno è caratterizzato dal tipo di accuratezza che si vuole ottenere dal risultato. Il vantaggio enorme è dunque il poter utilizzare delle teorie molto raffinate in quei punti critici della struttura, che possono coincidere con le regioni di concentrazione delle tensioni e/o le zone di applicazione dei carichi, mentre nelle zone meno critiche e non soggette a problematiche, possono essere utilizzate teorie meno raffinate. In tal maniera si può eseguire un'analisi strutturale sulla globalità della struttura, richiedendo un costo computazionale inferiore al computer ed ottenendo risultati molto accurati.
Il campo di spostamenti si può quindi riscrivere introducendo l'apice i alla funzione di espansione, riferito al nodo della struttura:

$$\mathbf{u}(x, y, z) = N_i(x, y) F_\tau^i(z) \mathbf{u}_{\tau i} \tag{2.61}$$

Dove $\tau = 1...M^i$, con M^i che serve ad indicare il fatto che il numero di termini di

espansione M può essere differente per ogni nodo, e $i = 1...N_n$, con N_n identificante il numero di nodi della struttura.
Quella presentata è la casistica della formulazione 2D in quanto quella che verrà utilizzata in questo lavoro di tesi, ma le altre formulazioni dimensionali possono essere ottenute con procedimento analogo.
Nello svolgimento di questo elaborato, l'approccio global/local segue quanto appena esposto e verrà utilizzato per l'individuazione della distribuzione delle teorie più efficienti da utilizzare in delle strutture di tipo guscio, affrontando diverse casistiche di spessore e condizioni di vincolo. L'analisi fornirà risultati non su ogni singolo nodo della mesh, ma su ogni elemento in cui è stata suddivisa la struttura. Pertanto, l'ordine del modello sarà differente per ogni elemento ed associato a tutti i nodi che lo compongono. Inoltre, si fa notare che nei nodi in comune tra più elementi, il software di calcolo utilizzato andrà a considerare per quei nodi l'ordine di teoria strutturale più elevato tra quelli definiti per gli elementi.

2.5 Metodo assiomatico-asintotico

Il metodo assiomatico-asintotico, dall'inglese *axiomatic-asymptotic method* (AAM), presentato in [14] ed integrato con la formulazione CUF, rappresenta un approccio molto utile per poter creare delle teorie ridotte di modelli strutturali che abbiano la stessa accuratezza della loro espansione completa, ma con il vantaggio di avere un costo computazionale inferiore. Il nome di questo metodo deriva dal fatto che vengono effettuate delle ipotesi di tipo assiomatico per ottenere dei risultati asintotici, andando a studiare l'influenza delle differenti variabili che compongono un modello.
La sua applicazione nello svolgimento di questa tesi è prevista per l'ottenimento di *Best Theory Diagram* (BTD) e sullo studio Global/Local, entrambi riferiti ad analisi dinamiche effettuate su diverse tipologie di strutture per geometria e condizioni di vincolo.

2.5.1 Best Theory Diagram

L'implementazione del metodo AAM nell'ottenimento dei BTD all'interno di questa tesi segue l'approccio utilizzato in [19] in cui l'ipotesi assiomatica alla base consiste nella scelta della teoria utilizzata per la risoluzione del problema strutturale.
I passi seguiti dal procedimento sono dunque i seguenti:

1. Vengono definiti i parametri del problema strutturale quali geometria, condizioni di vincolo, carichi e materiale.

2. Viene effettuata l'assunzione assiomatica per cui la teoria esatta di riferimento coincide con quella di espansione di Taylor di ordine quattro (N = 4).

3. Il metodo CUF viene utilizzato per generare le equazioni di governo della teoria scelta in precedenza.

4. Ad ogni analisi, la soluzione viene calcolata eliminando uno o più termini dell'espansione per tutte le possibili combinazioni coincidenti con 2^{12}, essendo 15 il numero di termini che compongono la teoria scelta, ma considerando i primi tre sempre presenti.

5. Ad ogni combinazione viene valutato un parametro di controllo per l'accuratezza, che nel caso delle analisi a vibrazioni libere che verranno effettuate successivamente, corrisponde con l'errore percentuale rispetto al valore di riferimento per ognuna delle prime dieci frequenze naturali del sistema.

Con questi passi si ottiene la distribuzione di termini attivi, per ogni numerosità, corrispondente alla teoria ridotta per cui si ottiene il minor errore nel calcolo della frequenza, rispetto alla teoria completa. La rappresentazione può avvenire sul piano cartesiano, riportando in ascissa l'errore, indicandolo solitamente con scala logaritmica, ed in ordinata i gradi di libertà (DOF) ovvero il numero di termini attivi. Questa rappresentazione grafica permette di osservare facilmente l'effetto che la numerosità di termini attivi produce sulla diminuzione dell'errore e quindi sul miglioramento dell'accuratezza del modello.
Oltre a quella sul piano cartesiano, viene utilizzata una rappresentazione grafica in tabella utile a dimostrare esattamente quali siano i termini attivi, riferiti ad una precisa numerosità, che forniscono il minor errore sul parametro di controllo. Considerando ad esempio le equazioni (2.57) per il campo di spostamenti per un modello 2D limitandosi all'ordine N = 4, la rappresentazione per la generica numerosità di dieci termini attivi è quanto riportata in tabella 2.5. L'indicazione se il termine è attivo oppure no, avviene con l'utilizzo di triangolo nero in caso di termine attivo, bianco in caso di termine inattivo.

Tabella 2.5: Rappresentazione posizione dei termini attivi.

DOF	$u_{\alpha 1}$	$u_{\beta 1}$	u_{z1}	$u_{\alpha 2}$	$u_{\beta 2}$	u_{z2}	$u_{\alpha 3}$	$u_{\beta 3}$	u_{z3}	$u_{\alpha 4}$	$u_{\beta 4}$	u_{z4}	$u_{\alpha 5}$	$u_{\beta 5}$	u_{z5}
10	▲	▲	▲	▲	▲	△	△	△	▲	▲	△	▲	▲	▲	△

2.5.2 Best Theory Distribution

L'approccio AAM viene utilizzato anche per lo studio global/local, andando a creare delle *Best Theory Distribution* per decretare il tipo di teoria utilizzato per ogni elemento e la sua distribuzione all'interno di una mesh strutturale. Il procedimento utilizzato è molto simile a quello visto in precedenza, in particolare quello adottato in questo lavoro segue i passi esposti in [20], che sono i seguenti:

1. Vengono definiti i parametri del problema strutturale quali geometria, condizioni di vincolo, carichi e materiale.

2. Viene effettuata l'assunzione assiomatica per cui la teoria esatta di riferimento coincide con quella di espansione di Taylor di ordine quattro (N = 4).

3. Viene definita la mesh FEM. L'applicazione in questa tesi avverrà su elementi di tipo piastra e guscio, con mesh definita da 4×4 elementi L9.

4. Il metodo CUF viene utilizzato per generare le equazioni di governo per la distribuzione di teorie considerata.

5. Ad ogni analisi, la soluzione viene calcolata sostituendo uno o più elementi FEM con una toeria di ordine inferiore, in questo caso FSDT, per tutte le possibili combinazioni. Trattandosi di una mesh composta da 16 elementi, le combinazioni totali risultano essere 2^{16}.

6. Ad ogni combinazione viene valutato un parametro di controllo per l'accuratezza, che nel caso di analisi a vibrazioni libere corrisponde con l'errore percentuale rispetto al valore di riferimento per ognuna delle prime dieci frequenze naturali del sistema.

In questa maniera si ottiene la distribuzione di teorie ottimale per ottenere un risultato accurato con un minor costo computazionale, essendo la teoria FSDT composta da meno termini rispetto a quella completa di espansione di Taylor del quarto ordine.
Anche in questo caso vengono proposte due tipologie di rappresentazione dei risultati, di cui la prima, quella sul piano cartesiano, risulta essere identica a quella già presentata se non con la differenza che il numero di termini attivi e quindi di DOF è in questo caso maggiore.
L'altro tipo di rappresentazione invece, analogamente a quanto visto in precedenza, viene utilizzata per dimostrare quali siano effetivamente gli elementi della mesh in cui è stata utilizzata una teoria piuttosto che l'altra. Viene utilizzato un grafico come quello di figura 2.8 in cui si può vedere la distribuzione degli elementi sulla mesh 4×4 della struttura colorati di nero nel caso venga utilizzata la teoria FSDT, altrimenti di colore bianco nel caso sia utilizzata la teoria di ordine maggiore con espansione di Taylor del quarto ordine (N = 4).

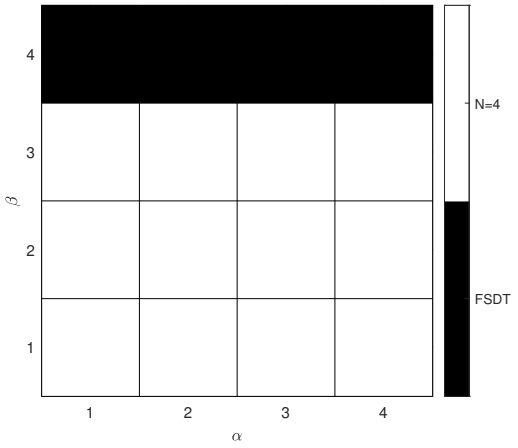


Figura 2.8: Rappresentazione Best Theory Distribution.

Capitolo 3

Reti neurali

Le reti neurali artificiali (ANN) rappresentano un'evoluzione del classico machine learning, richiedendo un minimo intervento da parte dell'uomo nell'ottenimento dei risultati. L'idea alla base del funzionamento di tali reti, è quella di creare degli algoritmi in grado di adattarsi e modificare la propria struttura interna in relazione alla funzione obiettivo richiesta [21]. Il loro campo di applicazione risulta essere molto ampio, ma si rendono particolarmente adatte per la risoluzione di problemi nonlineari, grazie alla loro capacità di adattarsi e ricostruire le relazioni che governano tali problemi.
La denominazione rete neurale deriva dall'analogia con i neuroni biologici, dei quali si vuole riprodurre il funzionamento di calcolo logico come analizzato ed esposto in [22]. Questa analogia tra neuroni biologici ed artificiali è molto forte, infatti si può schematizzare il funzionamento di un neurone biologico come riportato in figura 3.1. I dendriti si occupano di pesare i dati di input, forniti dall'esperienza reale della persona a cui appartengono e/o da altri neuroni, per poi trasmetterli al neurone, il quale li processa e li apprende, andando poi a creare nuove connessioni trasmettendo ciò che è stato elaborato tramite gli assoni.

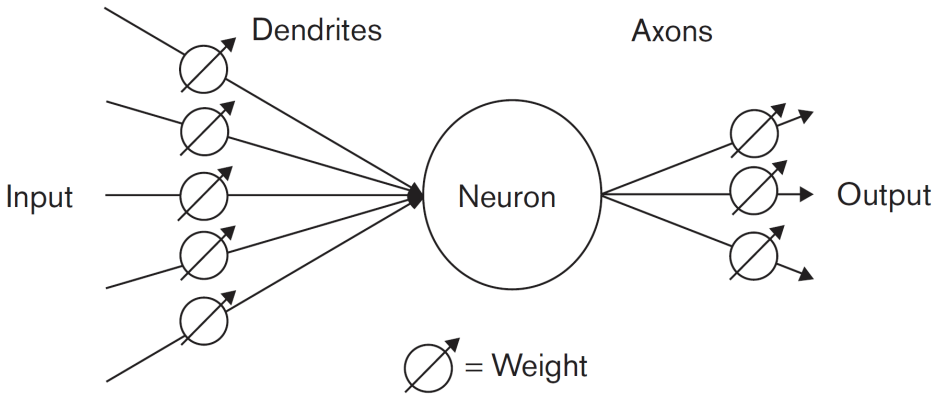


Figura 3.1: Schema funzionamento neurone biologico [21].

La rete neurale artificiale viene creata simulando lo stesso processo di apprendimento. Vengono creati dei layer di input, i quali contengono i dati iniziali da processare che verranno poi collegati a degli hidden layers, così chiamati perché normalmente non visibili all'utente, ma solo a chi ne costruisce la struttura. Ogni layer è composto da

un numero arbitrario di neuroni artificiali, i quali possono essere di vario tipo nel loro funzionamento. L'ultimo layer corrisponde con quello di output, in cui si ottengono i risultati impostati come obiettivo. Ognuno dei layer citati viene collegato tra loro in maniera univoca. Viene proposta una rapprenstazione schematica di rete neurale artificiale in figura 3.2.

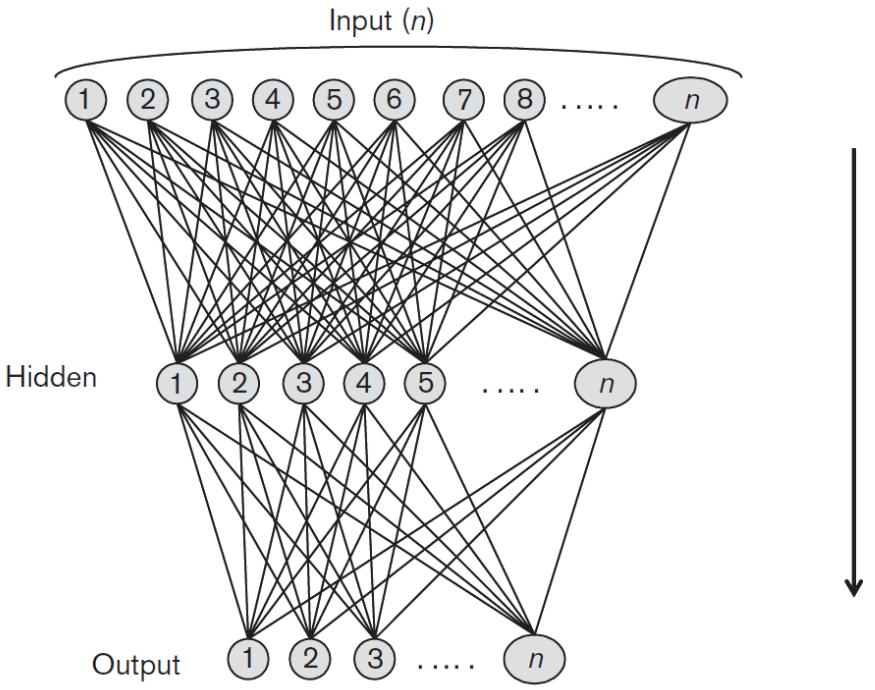


Figura 3.2: Schema funzionamento ANN [21].

L'elemento fondamentale alla base di tutto questo è proprio l'apprendimento. Si rende quindi necessario, nella costruzione della struttura di una rete neurale artificiale, il fornire dei dati comprensivi sia di input che di output per poter permettere all'algoritmo di imparare da essi la relazione che li lega. Inoltre, come nella vita delle persone, l'esperienza viene appresa dopo essere ripetuta, così anche accade per le ANN, dove si rende necessario ripetere più volte lo step di apprendimento, al fine di poter replicare e prevedere un risultato non ancora affrontato, grazie all'adattamento delle funzioni e dei parametri interni della rete.
Entrando nel dettaglio del modello matematico alla base dei singoli neuroni, si esprime la seguente funzione per il calcolo dell'output prodotto da un neurone a singolo input:

$$a = f(wp + b) \tag{3.1}$$

Dove a corrisponde all'ouput, p è l'input che viene moltiplicato per il peso w e sommato al termine di *bias* b. Il contenuto delle parentesi, che nella rappresentazione schematica di figura 3.3 coincide con il termine n, viene utilizzato come argomento all'interno di

una specifica *funzione di trasferimento* f, anche chiamata di *attivazione*. Nel caso semplice di neurone a singolo input, tutti questi termini sono degli scalari.

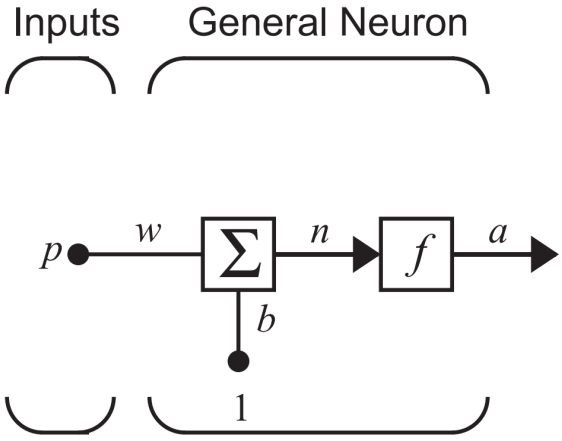


Figura 3.3: Schema neurone a singolo input [23].

La funzione di attivazione è scelta arbitrariamente, in base alle esigenze, nel momento in cui viene decisa l'architettura della rete. Alcune di queste, in particolare quelle di utilizzo nello svolgimento di questa tesi, verranno presentate successivamente. I termini w e b invece, sono quei parametri interni alla rete neurale che vanno a modificarsi nel processo di training della rete, adeguandosi ai risultati da produrre.
Tipicamente però, i neuroni artificiali che vengono utilizzati hanno più di un input. Nel gestire questa moltitudine, vengono utilizzati pesi con differenti valori per ognuno dei dati in entrata.
La (3.1) può dunque essere riscritta utilizzando una notazione matriciale, dove $\mathbf{W}$ rappresenta il vettore dei pesi e $\mathbf{p}$ coincide con il vettore degli R input.

$$a = f(\mathbf{W}\mathbf{p} + b) \tag{3.2}$$

Una rappresentazione schematica di un neurone con molteplici input è presentata in figura 3.4. Si osserva che nel definire i vari pesi è stata definita una notazione con due pedici, di cui il primo pedice indica il neurone di destinazione per il determinato peso, mentre il secondo pedice fornisce indicazione a riguardo della fonte a cui viene applicato.

Questa notazione è particolarmente utile perché nella realtà vengono utilizzate reti neurali in cui i vari layer sono composti da una moltitudine di neuroni e non da uno singolo. L'equazione (3.2) viene dunque rimodellata introducendo il pedice i per indicare il riferimento al singolo neurone appartenente all'insieme.

$$a_i = f_i(\mathbf{W}_i\mathbf{p}_i + b_i) \tag{3.3}$$

La struttura di un layer composto da S neuroni viene proposta in figura 3.5, mentre

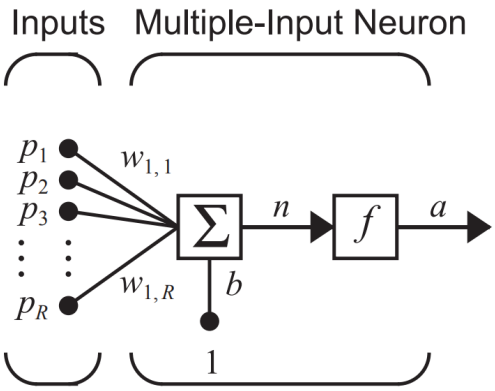


Figura 3.4: Schema neurone ad input multiplo [23].

in figura 3.6 è schematizzata l'espansione di quanto detto finora nel caso in cui siano presenti più di un layer di neuroni artificiali, comprensivo delle relazioni legate ad ogni singolo neurone. Andando a considerare gli output del secondo layer ed indicando con n^1 il numero di neuroni del layer precedente, quello che si ottiene è della forma:

$$a_i^2 = f_i^2(\sum_{j=1}^{n^1}(W_{i,j}^2(f^1(\mathbf{W}^1\mathbf{p} + \mathbf{b}^1))_j) + b_i^2) \tag{3.4}$$

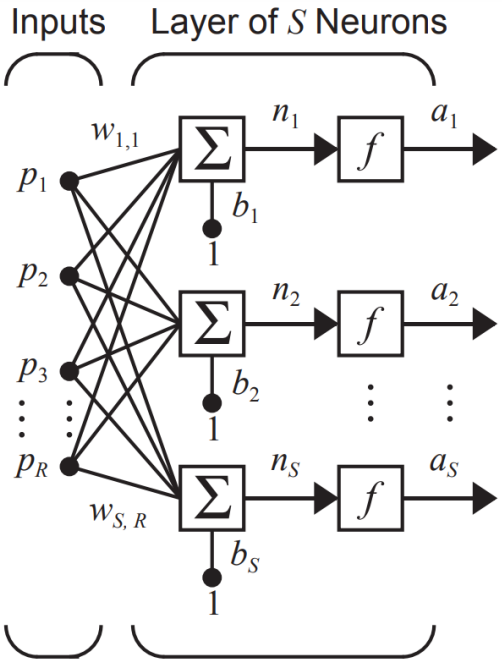


Figura 3.5: Schema layer di S neuroni [23].

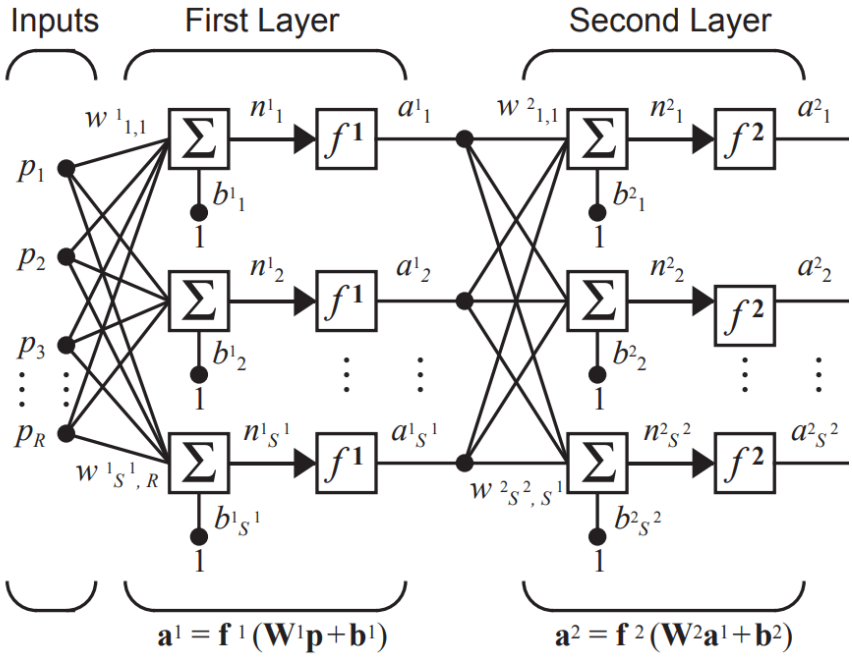


Figura 3.6: Schema rete neurale a due layer [23].

3.1 Funzioni di attivazione

Attualmente esistono numerose funzioni di attivazione implementabili all'interno delle reti neurali artificiali, ognuna con le proprie peculiarità che possono riguardare il dominio, la classe di continuità e la nonlinearità. La scelta di quali utilizzare viene effettuata in fase di progettazione dell'architettura della rete e va a considerare il tipo di problema da affrontare ed il tipo di dati a disposizione che verranno forniti alla rete. All'interno di questa tesi vengono presentate solamente le funzioni utilizzate nel suo svolgimento, per una consultazione più completa ci si può riferire a [23] e [24].

3.1.1 Rectified Linear Unit

La *Rectified Linear Unit* (ReLU) è una delle funzioni di attivazione più comunemente utilizzate nell'ambito delle moderne reti neurali. Viene definita tramite la relazione matematica:

$$y = max(0, x) \tag{3.5}$$

Si tratta di una funzione nonlineare, rappresentata nel grafico di figura 3.7, definita tra 0 e $+\infty$. L'inserimento di questa funzione di attivazione all'interno di un neurone, si occupa semplicemente di riportare in output lo stesso valore dell'input, ma solo se quest'ultimo è positivo. In caso contrario, fornisce come ouput un valore pari a zero. I vantaggi principali del suo utilizzo sono:

- L'implementazione è molto immediata trattandosi semplicemente di una funzione di ricerca del massimo valore. Non essendoci una matematica di base difficile da applicare, il costo computazionale così come il tempo necessario al training sono entrambi molto bassi.
- Il comportamento quasi lineare è più semplice da ottimizzare e porta ad una convergenza dei risultati più velocemente.
- Poter fornire molti zero in output è un vantaggio per creare un modello che funzioni con una certa sparsità, attivando solo i neuroni effettivamente necessari e diminuendo ulteriormente il tempo di calcolo.

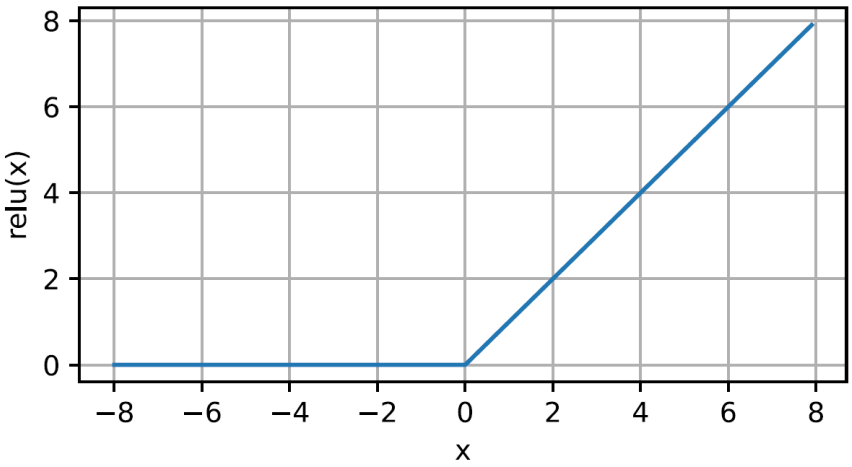


Figura 3.7: Funzione ReLU [24].

3.1.2 Sigmoid

La funzione *Sigmoid* è anch'essa una funzione nonlineare, ma fornisce risultati in un intervallo compreso tra 0 e 1. Questo si rivela particolarmente utile quando si vuole ottenere dei dati oppurtunamente normalizzati o comunque ristretti in una certa distribuzione.
La funzione è definita da:

$$\sigma(x) = \frac{1}{1 + e^{-x}} \tag{3.6}$$

La sua implementazione avviene maggiormente nei layer di output delle reti neurali in cui si vuole ottenere un risultato che possa essere una probabilità o una percentuale. L'andamento della funzione viene riportato nel grafico in figura 3.8.

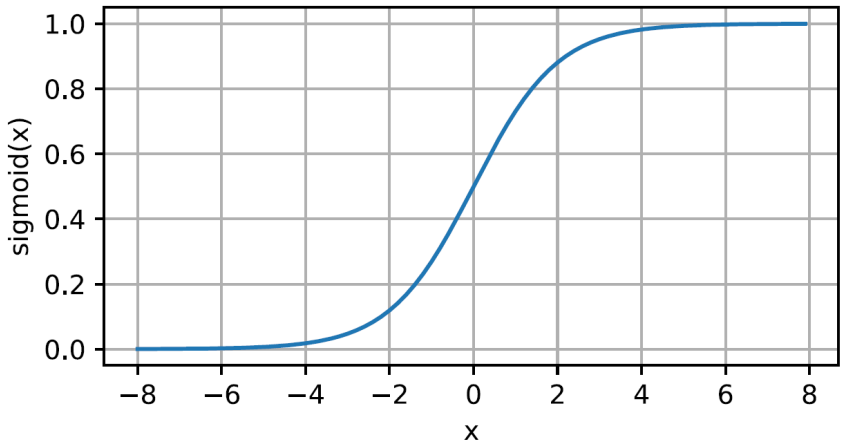


Figura 3.8: Funzione Sigmoid [24].

3.2 Training della rete

Si passa ora a descrivere il processo di training della rete neurale, ovvero di come questa effettui l'aggiornamento dei parametri visti in precedenza, al fine di ottenere delle relazioni tra dati in input e dati di output. Come esposto in [24], esistono numerose metodologie per effettuare il training di una ANN, ognuna con il proprio campo di applicazione. Nell'interesse di quanto viene svolto in questa tesi, si va ad esaminare il processo di training identificato come *Backpropagation*. Sinteticamente consiste nell'eseguire l'algoritmo, valutare un errore rispetto a dei dati di validazione, non inclusi nel gruppo di dati di training, per poi andare a ritroso nell'algoritmo modificando i vari parametri in funzione dell'errore che si era valutato.
Nel dettaglio, il metodo calcola il gradiente dei parametri della rete neurale procedendo in ordine inverso, iniziando dal layer di output e concludendo con il layer di input, senza trascurare alcuno degli hidden layers, seguendo la regola della catena nella determinazione delle derivate. Tale regola, supponendo di avere uno scalare $z \in \mathbb{R}$ funzione di un vettore $\mathbf{y} \in \mathbb{R}^H$, funzione a sua volta di un altro vettore $\mathbf{x} \in \mathbb{R}^W$, esprime la derivata:

$$\frac{\partial z}{\partial \mathbf{x}^T} = \frac{\partial z}{\partial \mathbf{y}^T} \cdot \frac{\partial \mathbf{y}}{\partial \mathbf{x}^T} \tag{3.7}$$

Questa relazione viene applicata per calcolare il gradiente di ogni parametro e variabile intermedio, partendo dal gradiente della funzione obiettivo rispetto ad una funzione di calcolo dell'errore scelta arbitrariamente. Riprendendo come esempio quanto presentato in [25], si considera la seguente *loss function* per il calcolo dell'errore:

$$z = \frac{1}{2}||t - x^L||^2 \tag{3.8}$$

Dove t rappresenta il parametro target che si vuole ottenere mediante l'utilizzo della

rete neurale, mentre x^L è proprio il valore ottenuto dalla prediction.
Introducendo il parametro k, relativo al numero dell'iterazione, ed il parametro η denominato *learning rate* e tipicamente impostato ad un valore di 0.001, i vari pesi e termini di bias di ogni layer i-esimo vengono aggiornati secondo le relazioni:

$$(w^i)^{t+1} = (w^i)^t - \eta \frac{\partial z}{\partial (w^i)^t} \tag{3.9}$$

$$(b^i)^{t+1} = (b^i)^t - \eta \frac{\partial z}{\partial (b^i)^t} \tag{3.10}$$

Sviluppando la derivata parziale presente in queste due equazioni utilizzando la relazione (3.7), si può scrivere:

$$\frac{\partial z}{\partial (w^i)} = \frac{\partial z}{\partial (x^i)} \cdot \frac{\partial (x^i)}{\partial (w^i)} \tag{3.11}$$

In cui x^i è l'input fornito al layer, coincidente con l'output a^{i-1} del layer precedente.
Per semplicità è stato riportato solo lo sviluppo della (3.9), essendo il metodo per la (3.10) analogo.
Ad ogni layer vengono dunque calcolati due set di gradienti differenti:

- Il gradiente $\frac{\partial z}{\partial (w^i)}$, che viene utilizzato per aggiornare i parametri del layer attuale i.
- Il gradiente $\frac{\partial z}{\partial (x^i)}$, che viene utilizzato per aggiornare i parametri del layer precedente $(i-1)$.

Questo procedimento viene effettuato per ogni layer, fino a quello di input. Quando avviene il completamento di un ciclo di aggiornamento dei parametri utilizzando tutti i dati di training, ci si riferisce al procedimento utilizzando il termine *epoch*. In generale viene impostato arbitrariamente un numero di *epochs* con cui la rete andrà ad effettuare il training. Aumentare il numero di epoche di training serve a far ridurre in continuazione l'errore sul set di training valutato tramite la funzione apposita, ma serve prestare attenzione in quanto può verificarsi che l'errore di validazione, oltre a raggiungere un punto in cui non si abbassa ulteriormente, potrebbe riprendere ad aumentare. Se ciò dovesse succedere, si rende necessario diminuire il parametro di epochs, in quanto la rete è in *overfitting*, ovvero ha memorizzato i dati che sono stati forniti durante la fase di training ed è solamente in grado di riprodurre quelli, ma non di fornire prediction su set di dati con i quali non è stata allenata.
Inoltre, un'altra strategia utilizzata all'interno della fase di training è quella di definire un ulteriore parametro, denominato *batch size*, che serve a suddividere in gruppi i dati di input, in modo da far compiere alla rete neurale un aggiornamento dei propri parametri interni dopo aver analizzato ogni mini-batch. Questo comporta che all'interno di una singola epoca di training, la ANN si stia già adattando, permettendo l'utilizzo di un numero minore di epoche per arrivare a convergenza.

3.3 Tipologia di layer

Attualmente esistono differenti tipologie di layer che sono stati sviluppati con lo studio sempre più esteso del machine learning. Ciascuno di questi presenta le proprie peculiarità, sia di algoritmo che di funzionamento stesso del neurone artificiale, ed è applicato in funzione del tipo di problema che si vuole andare ad affrontare. In letteratura sono presenti diversi studi a riguardo, come in [23], [26] e [24]. In questa tesi si riportano solamente le tipologie di layer utilizzate durante lo svolgimento per la risoluzione dei problemi di interesse.

3.3.1 Fully connected

Un *fully connected* layer, anche chiamato *dense*, viene tipicamente utilizzato negli ultimi layer di una rete neurale artificiale ai fini di migliorare i risultati ottenuti [27].
Un layer di questo tipo è definito tale in quanto ogni elemento di output viene elaborato sfruttando tutti i singoli elementi di input, rendedolo appunto profondamente connesso con il layer precedente.
I parametri tipici di questo layer, che verranno implementati, riguardano la dimensione, ovvero il numero di neuroni, e la funzione di attivazione, scelta tra quelle presentate precedentemente.

3.3.2 Convolutional

I layer di tipo *convolutional* sono ciò che sta alla base del funzionamento delle reti neurali convoluzionali [24]. Esistono differenti layer di questo tipo a seconda della dimensione (1D, 2D o 3D) su cui avviene la convoluzione. All'interno di questo progetto di tesi viene utilizzato il layer 2D, ovvero la convoluzione spaziale, pertanto verrà presentata solamente questa tipologia.
L'operazione di convoluzione consiste nella moltiplicazione dell'input, ovvero gli elementi di una matrice o un tensore di ordine superiore, per un set di pesi disposti in forma matriciale. Questa matrice di pesi prende il nome di *nucleo*, anche chiamato *filtro* od in inglese *kernel*. Durante l'operazione di convoluzione, il nucleo viene spostato sulla totalità dei dati di input da sinistra verso destra e dall'alto verso il basso di un passo definito arbitrariamente e, ad ogni passo, viene effettuatta la moltiplicazione per i pesi e la successiva somma per ottenere il valore di output. Un esempio di operazione di convoluzione con utilizzo di un nucleo 2x2 e passo unitario è riportato in figura 3.9.

Un esempio numerico della prima operazione di convoluzione eseguita sullo spazio in alto a sinistra dei dati di input, sfruttando i valori riportati in figura è la seguente:

$$1 * 1 + 2 * 1 + 4 * 1 + 5 * 1 = 12$$

Si può notare come l'output di questa operazione sia di dimensioni inferiori rispetto a quelle di input, fintanto che il nucleo utilizzato sia più grande di 1x1. Talvolta però, si rende necessario ottenere un output che abbia le stesse dimensioni di partenza, pertanto viene introdotto il *padding*, ovvero l'aumento fittizio delle dimensioni dell'input

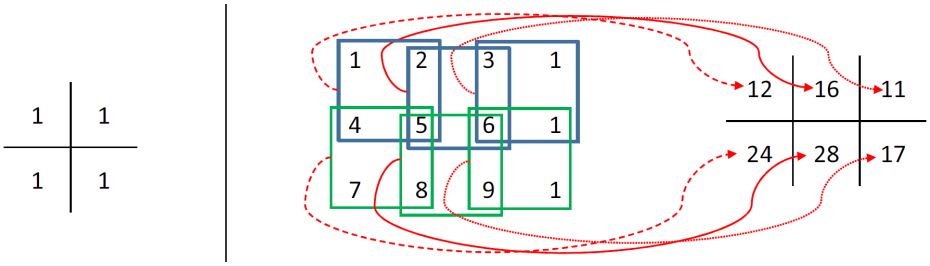


Figura 3.9: Operazione di convoluzione [25].

tramite l'aggiunta di righe e colonne di elementi nulli. Ciò permette di mantenere le dimensioni dell'input a valle della convoluzione.
I paramentri fondamentali da definire nella creazione di un layer di tipo convolutional sono quindi il numero di neuroni, la dimensione del nucleo, il passo (in inglese *stride*) che il nucleo andrà a compiere in ogni direzione, la presenza di padding, e la funzione di attivazione scelta.

3.3.3 Flatten

Questo tipo di layer, risulta essere molto più semplice rispetto ai precedenti, ma di importanza fondamentale soprattutto nell'implementazione di reti neurali in cui siano presenti diverse tipologie di layers. La sua funzione è quella di trasformare la forma del suo input andando a creare un unico vettore di dati come ouput. Diventa intuitivo capire come il suo utilizzo sia necessario in situazioni nelle quali, ad esempio, si desidera implementare un layer di tipo dense, che richiede in input un vettore, a seguito di un layer di tipo convolutional, il quale fornisce in output una matrice o tensore di ordine superiore.

Capitolo 4

Risultati preliminari

4.1 Analisi meccaniche

4.1.1 Analisi statiche trave

In questa prima parte della tesi sono state svolte diverse analisi statiche strutturali mediante software apposito, con l'intento di verificare i risultati ottenuti utilizzando diverse discretizzazioni. Ad ogni analisi sono stati quindi variati sia il numero, che il tipo di elementi utilizzati.

La trave considerata, rappresentata in figura 4.1, ha dimensioni $b = 3$ mm, $h = 5$ mm, $L = 30$ mm ed è soggetta ad un vincolo di tipo incastro sulla sezione d'estremità sinistra, mentre sulla sezione all'estremo opposto viene applicato uno spostamento di 2 mm. Viene utilizzato un materiale ortotropo con caratteristiche meccaniche: modulo di Young $E_1 = 12$ GPa, $E_2 = E_3 = 5$ GPa, coefficiente di Poisson $\nu = 0.2$, modulo a taglio $G = 2$ GPa. La laminazione prevede angolo di rotazione nullo rispetto agli assi y e z.

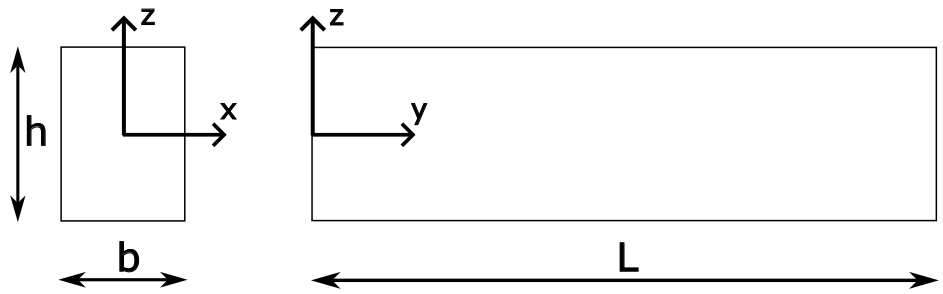


Figura 4.1: Rappresentazione geometria della trave.

Il primo obiettivo consisteva nel valutare la deformata della struttura estrapolando i valori dello spostamento verticale in alcuni punti prestabiliti, ripetere il procedimento per differenti discretizzazioni e riportare gli andamenti in un grafico.

La tipologia di mesh è costituita da elementi FEM lineari lungo l'asse y della trave, i quali possono avere, nel caso in esame, due o quattro nodi. Nel caso di due nodi

si identifica l'elemento con la sigla B2, nel caso di quattro nodi la sigla è B4.
Sulla sezione è invece presente una mesh refined, ovvero espansa, ottenuta in questo caso tramite espansione di Lagrange. La tipologia di elementi CUF bidimensionali presi in esame possono essere a quattro o a nove nodi, identificati rispettivamente dalle sigle L4 e L9.

La rappresentazione grafica degli elementi è riportata in figura 4.2 per quanto riguarda gli elementi monodimensionali utilizzati lungo l'asse y, mentre in figura 4.3 sono riportati gli elementi bidimensionali utilizzati nella sezione.

(a) Elemento B2. (b) Elemento B4.

Figura 4.2: Elementi 1D.

(a) Elemento L4. (b) Elemento L9.

Figura 4.3: Elementi 2D.

Il numero di gradi di libertà (DOF) è stato poi calcolato moltiplicando tra loro il numero di nodi lungo y, il numero di nodi sulla sezione e i 3 tre gradi di libertà di ogni singolo nodo. I risultati ottenuti con le varie combinazioni di discretizzazioni lungo y e sulla sezione xz vengono riportati in tabella 4.1. In particolare, sono stati riportati i valori di spostamento verticale u_z del punto situato a $[0, L/2, 0]$, presentando come confronto il valore ottenuto analiticamente utilizzando la teoria di Eluero-Bernoulli per cui $u_{z_b} = \frac{F_z L^3}{3EI}$. Inoltre, si specifica che l'utilizzo di due elementi L9 sulla sezione è stato fatto considerando i due elementi affiancati in verticale.

Gli gli andamenti degli spostamenti lungo l'asse della trave, rappresentativi della deformata della struttura sono stati rappresentati tramite Matlab e riportati in figura 4.4.

Tabella 4.1: Spostamento verticale u_z del punto $[0, L/2, 0]$.

El. FEM	El. CUF	DOF	$u_z[mm]$
			$u_{z_b} = 0.625$mm
1B2	1L4	24	1.0000
5B2	1L4	72	0.6451
2B2	1L9	81	0.6308
5B2	1L9	162	0.6445
6B4	1L4	228	0.6304
3B4	1L9	270	0.6297
5B2	2L9	270	0.6449
15B2	1L9	432	0.6315
5B2	4L9	450	0.6450
6B4	1L9	513	0.6300
9B4	1L9	756	0.6301
6B4	2L9	855	0.6310
6B4	4L9	1425	0.6310

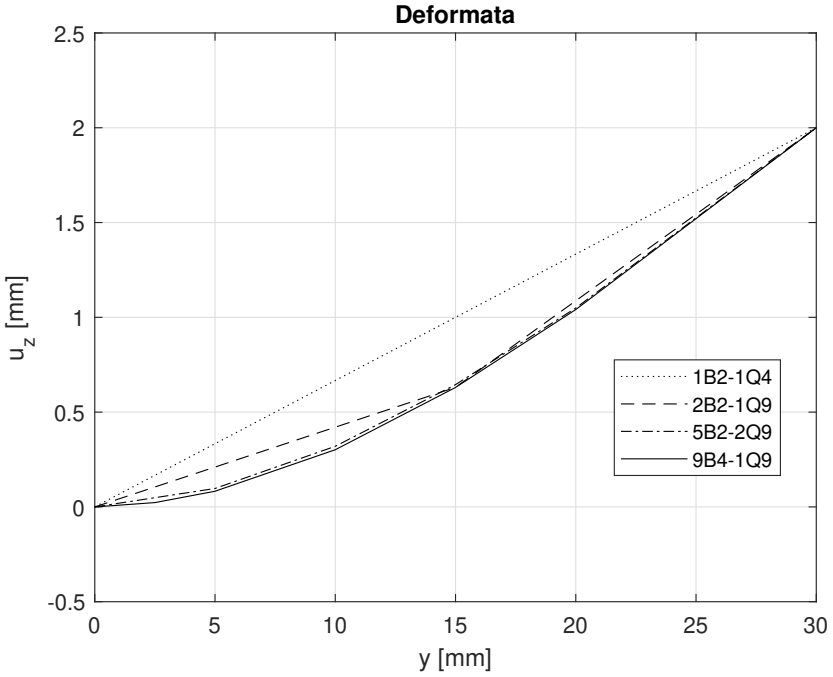


Figura 4.4: Deformata con diverse discretizzazioni.

Si può notare come la rappresentazione della deformata ottenuta dalla combinazione 1B2-1L4 sia quella che si discosta maggiormente dalla realtà. Questo è dovuto al fatto che il software aveva a disposizione solamente i nodi all'estremità della trave in cui calcolare gli spostamenti, pertanto quello che avviene nel mezzo è stato ottenuto mediante approssimazione lineare.
Infittendo la mesh, e quindi anche i nodi che rappresentano la struttura, il risultato di-

venta sempre più raffinato ed uniforme, tanto da rendere difficile distinguere i risultati appartenenti a differenti discretizzazioni.

Osservando nel dettaglio i valori dello spostamento verticale del punto $[0, L/2, 0]$ in funzione del numero di gradi di libertà, quello che si osserva è una rappresentazione quasi uniforme, indice del fatto che nella struttura presa in esame, la convergenza avviene molto rapidamente anche utilizzando un basso numero di nodi. L'unico valore che si discosta di molto dagli altri è quello del caso in cui viene utilizzato un unico elemento B2, proprio perché gli unici nodi lungo l'asse y sono quelli ai due estremi, pertanto il risultato fornito dal software di analisi è una media tra i valori conosciuti ai due estremi e non rappresenta il reale andamento della deformazione.

4.1.2 Replica risultati paper

In questa sezione si vuole andare a riprodurre alcuni dei risultati di analisi statiche riportati in [28] tramite il software di calcolo utilizzato finora, in modo da poterne confermare la sua validità. Per tutte le casistiche a seguire viene utilizzato lo stesso materiale isotropo di proprietà: modulo di Young E = 75 GPa, coefficiente di Poisson $\nu = 0.33$.

Sezione rettangolare

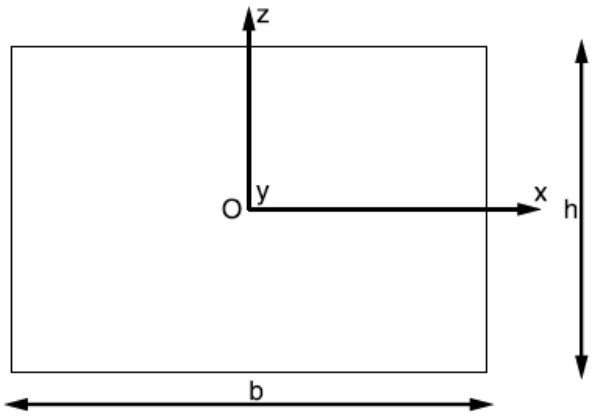


Figura 4.5: Sezione rettangolare.

Il primo esempio riprodotto consiste in una trave a sabalzo a sezione rettangolare, che si può vedere riprodotta in figura 4.5. Le dimensioni della sezione sono $b = h = 0.2$ m. Per la lunghezza della trave vengono considerati due differenti rapporti di snellezza pari a $L/h = 100$ e $L/h = 10$ corrispondenti rispettivamente ad una lunghezza $L = 20$ m ed $L = 2$ m. Alla trave è stato applicato un carico F_z di intensità -50 N nel punto di applicazione situato a [0, L, 0].
Il tipo di analisi statiche effettuate prevede l'utilizzo di diverse discretizzazioni lungo

l'asse della trave e per l'espansione sulla sezione, con lo scopo di valutare lo spostamento u_z nello stesso punto di applicazione del carico. Sono stati considerati tre differenti numeri di elementi lungo y, coincidenti con 5, 10 e 40 B4 e due differenti espansioni per la sezione, utilizzando inizialmente un elemento Q4 e poi un elemento Q9. Gli elementi coincidono con quelli riportati precedentemente in figura 4.2 e 4.3. La scelta della tipologia di elementi utilizzati lungo l'asse della trave è stata effettuata in seguito ad una valutazione per cui provando ad utilizzare elementi B2, il risultato ottenuto non era sufficientemente accurato. Questo era dovuto alla bassa numerosità di punti presenti sulla lunghezza della trave. L'utilizzo dei B4 ha invece portato dei risultati soddisfacenti e che vengono riportati in tabella 4.2 per la trave snella, ed in tabella 4.3 per la trave corta.

Tabella 4.2: Spostamento verticale u_z trave a sezione rettangolare. $L/h = 100$.

N. Elem.	5	10	40
$u_z \times 10^2$ $[m]$			
1 L4	-1.106	-1.111	-1.114
1 L9	-1.311	-1.323	-1.331
1 L4 [28]	-1.107	-1.111	-1.115
1 L9 [28]	-1.314	-1.324	-1.331

Tabella 4.3: Spostamento verticale u_z trave a sezione rettangolare. $L/h = 10$.

N. Elem.	5	10	40
$u_z \times 10^5$ $[m]$			
1 L4	-1.112	-1.115	-1.115
1 L9	-1.320	-1.327	-1.329
1 L4 [28]	-1.112	-1.115	-1.116
1 L9 [28]	-1.320	-1.327	-1.329

Si può notare come i risultati ottenuti per entrambe le travi siano molto simili a quelli di [28], con una differenza in difetto che si assottiglia all'aumentare del numero di elementi utilizzati nella discretizzazione.
Successivamente, è stato svolto un confronto anche sui risultati ottenuti dalla trave snella utilizzando espansioni sulla sezione del tipo 2x1 L4, 1x2 L4, 2x2 L4, dove la prima rappresenta l'utilizzo di due elementi L4 lungo l'asse x, la seconda rappresenta l'utilizzo di due elementi L4 lungo l'asse z, l'ultima rappresenta l'utilizzo di due elementi L4 sia lungo l'asse x sia lungo l'asse z. I risultati così ottenuti sono stati riportati in tabella 4.4.

Anche in questo caso si è dimostrata l'affidabilità dei risultati ottenuti dal software, che migliora continuamente all'aumentare del numero di elementi utilizzati nella discretizzazione.

Tabella 4.4: Effetto di diverse discretizzazioni sullo spostamento verticale u_z trave a sezione rettangolare. $L/h = 10$.

N. Elem.	1 L4	2x1 L4	1x2 L4	2x2 L4
$u_z \times 10^2$ $[m]$				
5	-1.106	-1.154	-1.220	-1.252
10	-1.111	-1.160	-1.228	-1.261
40	-1.114	-1.164	-1.234	-1.267
5 [28]	-1.107	-1.155	-1.222	-1.254
10 [28]	-1.111	-1.160	-1.229	-1.262
40 [28]	-1.115	-1.164	-1.234	-1.268

Sezione a C

In questo secondo esempio viene proposta nuovamente una trave a sbalzo, ma con sezione a C come riportato in figura 4.6. Le dimensioni della sezione sono $h = b_2 = 1$ m, $b_1 = b_2/2 = 0.5$ m, rapporto di snellezza $L/h = 20$ per una lunghezza totale della trave $L = 20$ m, rapporto $h/t = 10$ per uno spessore uniforme $t = 0.1$ m. Sono applicati due carichi di intensità -1 N e $+1$ N in direzione z, rispettivamente nei punti [0, L, +0.4] e [0, L, -0.4].
Si vuole valutare l'entità dello spostamento verticale u_z riferito al punto del lembo libero superiore situato a [-0.5, L, +0.4]. Vengono replicate le discretizzazioni presenti in [28], utilizzando 10 elementi L4 lungo l'asse della trave, combinati inizialmente con la discretizzazione sulla sezione avente 6 elementi L9 e, successivamente con quella avente 9 L9. Entrambe le discretizzazioni vengono riportate in figura 4.7. Viene inoltre proposto un risultato ottenuto con l'utilizzo di elementi L4 in sostituzione ai L9.

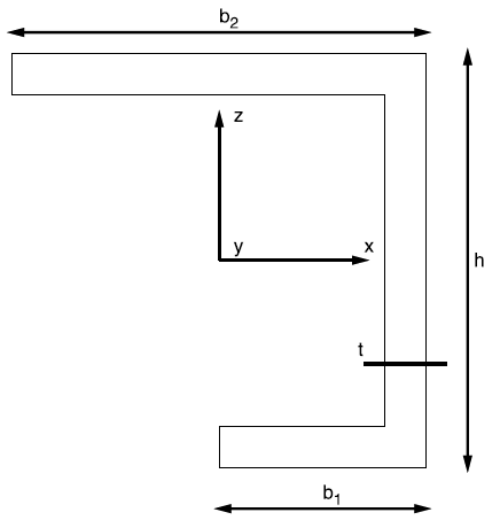
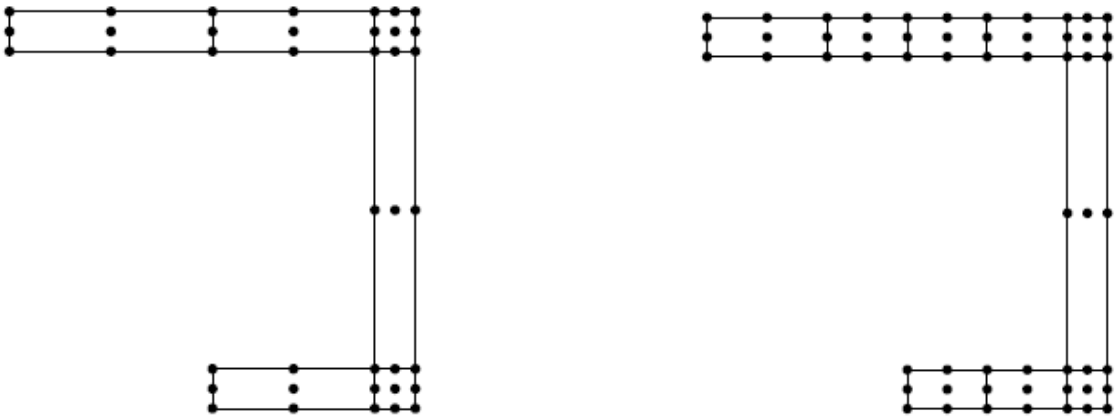


Figura 4.6: Sezione a C.



(a) Sezione con 6 elementi L9.

(b) Sezione con 9 elementi L9.

Figura 4.7: Discretizzazioni sezione a C.

I risultati ottenuti sono riportati nella tabella 4.5, nella quale è anche riportato, nella seconda colonna, il numero di gradi di libertà che compone la struttura, ottenuto moltiplicando tra loro il numero di nodi della sezione, il numero di nodi lungo l'asse y e le tre direzioni x, y, z.

Tabella 4.5: Spostamento verticale u_z trave sezione a C.

	DOF	$u_z \times 10^8$ $[m]$
6 L4	1302	-0.873
9 L4	1860	-1.050
6 L9	3627	-2.904
9 L9	5301	-2.957
6 L9 [28]	3627	-2.930
9 L9 [28]	5301	-2.982

I risultati ottenuti risultano essere simili a quelli già verificati in [28] e sono pertanto attendibili. Questo ci fornisce indicazione su come il risultato ottenuto utilizzando una discretizzazione con elementi L4 sulla sezione sia molto distante dalla convergenza, rendendo il modello così impostato non utilizzabile per analisi future.

Sezione quadrata aperta

L'ultimo esempio riportato consiste in una trave a sbalzo con sezione quadrata cava e aperta, come rappresentato in figura 4.8. Le dimensioni della sezione sono $b = h = 1$ m, rapporto di snellezza $L/h = 20$ per una lunghezza totale della trave $L = 20$ m, rapporto $h/t = 10$ per uno spessore uniforme $t = 0.1$ m. L'apertura sulla base è stata identificata imponendo uno spazio arbitrario di 0.01 mm tra i due lembi. Vengono applicate due forze di intensità -1 N e $+1$ N in direzione x, nei punti [-5∗10^{-6}, L,

-0.45] e [$5*10^{-6}$, L, -0.45].
Si vuole valutare lo spostamento orizzontale u_x dei punti situati a [$\pm 5*10^{-6}$, L, $-h/2$]. La trave viene discretizzata con l'utilizzo di 10 elementi B4 lungo l'asse, mentre sulla sezione vengono considerate tre differenti discretizzazioni, riportate in figura 4.9 e coincidente con quelle presentate in [28]. In particolare, il primo caso prevede la distribuzione di nove elementi L9, mentre i due casi successivi utilizzano undici elementi L9 infittendo la discretizzazione inizialmente sul lato inferiore della sezione e successivamente sui due lati verticali.

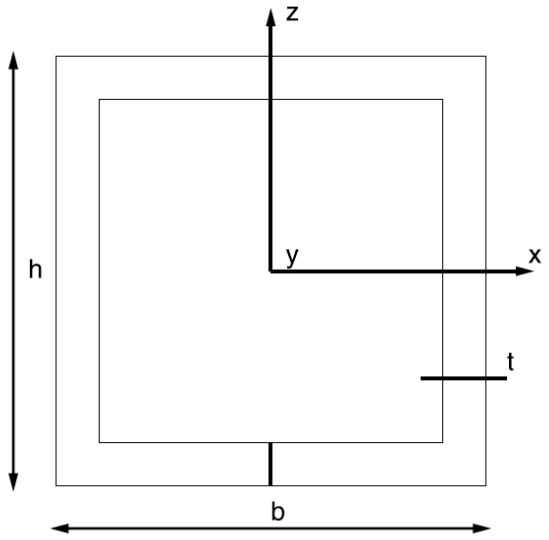
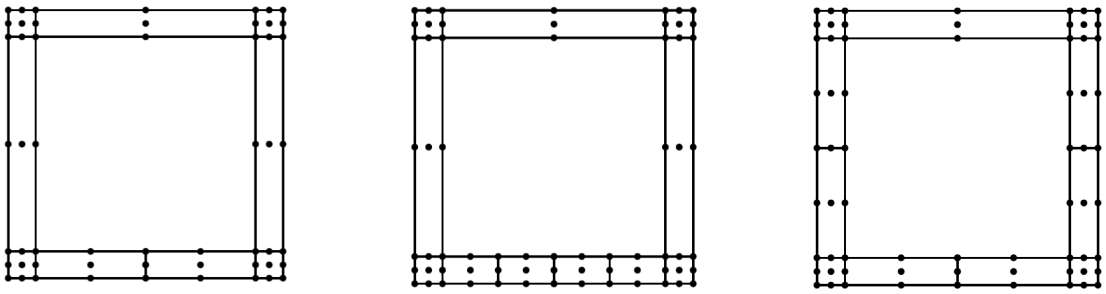


Figura 4.8: Sezione quadrata cava aperta.



(a) 9 elementi L9. (b) 11 elementi L9^a. (c) 11 elementi L9^b.

Figura 4.9: Discretizzazioni sezione quadrata cava aperta.

I risultati ottenuti vengono riportati in tabella 4.6, associati al numero di gradi di libertà che compongono l'intera struttura nei tre differenti casi di discretizzazione.
Anche in questo caso i risultati ottenuti sono in linea con quelli già verificati in [28], rendendo così il metodo attendibile. Una parte della discrepanza dal risultato esatto è

dovuto anche all'aver considerato un'apertura arbitraria di 0.01 mm, al fine di rendere possibile il calcolo tramite il software.

Tabella 4.6: Spostamento u_x trave sezione quadrata cava aperta.

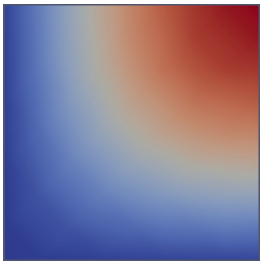
	DOF	$u_x \times 10^8$ $[m]$
9 L9	5301	4.860
11 L9^a	6417	4.870
11 L9^b	6417	5.097
9 L9 [28]	5301	4.884
11 L9^a [28]	6417	4.888
11 L9^b [28]	6417	5.116

4.1.3 Analisi vibrazioni libere

Lo stesso software di calcolo verrà ampiamente utilizzato nello svolgimento di questa tesi anche nell'ambito di analisi dinamiche, in particolare l'analisi delle vibrazioni libere, per l'individuazione delle frequenze proprie delle varie strutture in esame. Nell'utilizzo di questa applicazione vengono proposte strutture di tipo piastra e guscio, nelle quali la discretizzazione viene impostata seguendo i principi della CUF, utilizzando dunque elementi bidimensionali sul piano delle facce ed elementi monodimensionali lungo lo spessore allineato con l'asse z. In questa fase di training nell'utilizzo del codice di calcolo viene considerata, con le opportune condizioni di simmetria, un quarto di un guscio di dimensioni $a = b = 1$ m, $R_\alpha = R_\beta = R$, $R/a = 20$, $a/h = 100$. La laminazione utilizzata è 0/90, mentre le condizioni di vincolo sono quelle del singolo appoggio su tutti i quattro lati. Le proprietà del materiale sono $E_1/E_2 = 25,\ G_{12}/E_2 = G_{13}/E_2 = 0.5,\ G_{13}/E_2 = 0.2,\ \nu = 0.25$.
In tabella 4.7 vengono riportati i valori delle prime dieci frequenze naturali valutate dal codice di calcolo. Inoltre, a seguire, vengono proposte le rappresentazione delle forme modali ad esse associate.

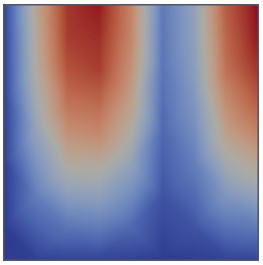
Tabella 4.7: Valori delle prime dieci frequenze.

	f [Hz]
Frequenza I	3.627×10^{-4}
Frequenza II	1.775×10^{-3}
Frequenza III	1.783×10^{-3}
Frequenza IV	2.669×10^{-3}
Frequenza V	4.827×10^{-3}
Frequenza VI	4.842×10^{-3}
Frequenza VII	5.411×10^{-3}
Frequenza VIII	5.415×10^{-3}
Frequenza IX	7.446×10^{-3}
Frequenza X	9.819×10^{-3}

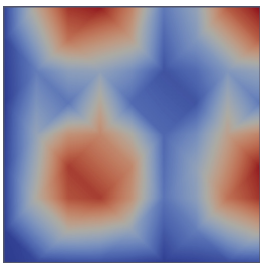


(a) Frequenza I.

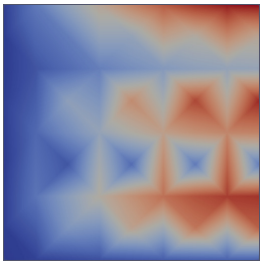
(b) Frequenza II.



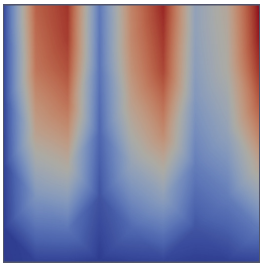
(c) Frequenza III.



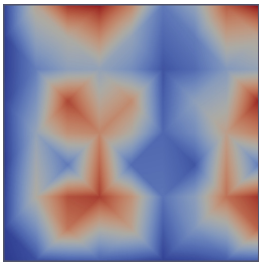
(d) Frequenza IV.



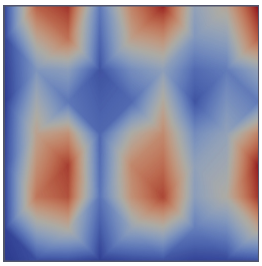
(a) Frequenza V.



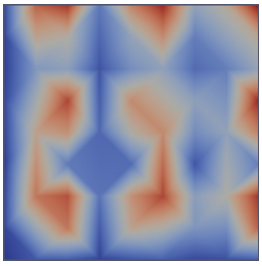
(b) Frequenza VI.



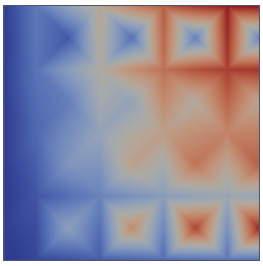
(c) Frequenza VII.



(d) Frequenza VIII.



(a) Frequenza IX.



(b) Frequenza X.

4.2 Testing reti neurali

4.2.1 Best theory diagrams

L'utilizzo della programmazione in Python risulta essere molto utile nell'analisi dei dati, soprattuto quando questi raggiungono una numerosità tale da essere improponibile non ricorrere all'aiuto di un calcolatore.
Una prima utile implementazione è stata quella di scrivere un codice che fosse in grado di fornire in ouput un Best Theory Diagram (BTD), ovvero un file contentente informazioni riguardanti la miglior scelta di termini attivi per ogni ordine di teoria utilizzata, in seguito all'inserimento di un opportuno file di dati. Il file utilizzato per l'input riguarda i risultati di diverse analisi statiche effettuate su una struttura impiegando diverse combinazioni di termini attivi per definire l'ordine della teoria utilizzata. La sua struttura prevede la presenza di: nella prima colonna un indice progressivo; nella

seconda colonna viene indicato il numero di termini attivi della teoria con cui viene effettuato il calcolo di analisi statica; nella terza colonna è presente l'errore percentuale sul risultato rispetto al risultato di riferimento ottenuto con tutti i 12 termini attivi; nelle successive colonne sono presenti le diverse combinazioni di termini identificando con 0 quelli inattivi e con 1 quelli attivi della teoria, facendo riferimento all'ordine crescente dal primo al quarto grado rispettivamente per le coordinate x, y, z. All'interno di questo file sono presenti 4096 diverse combinazioni. Lo scopo è quello di ottenere un file contenente indicazioni su quali siano i termini attivi, per ogni numerosità, in grado di fornire il minor errore percentuale sul risultato strutturale. Per ottenere ciò, la scrittura del codice Python ha previsto l'implementazione di un doppio ciclo for per l'analisi di ogni riga procedendo in ordine crescente di numerosità di cui alla seconda colonna, con al suo interno un condizionale per la ricerca del minor errore percentuale. Il ciclo ha quindi fornito un vettore contenente gli indici delle righe, riferiti al file di input, nelle quali è presente la combinazione di termini attivi che produce il minore errore possibile. Il file di output, opportunamente rappresentato in tabella 4.8, presenta quindi nella prima colonna il numero di termini attivi, e nelle colonne successive la combinazione di termini attivi ed inattivi che permettono di ottenere il minor errore possibile.

Tabella 4.8: Output script Python - BTD.

DOF	$u_{\alpha 1}$	$u_{\beta 1}$	u_{z1}	$u_{\alpha 2}$	$u_{\beta 2}$	u_{z2}	$u_{\alpha 3}$	$u_{\beta 3}$	u_{z3}	$u_{\alpha 4}$	$u_{\beta 4}$	u_{z4}	$u_{\alpha 5}$	$u_{\beta 5}$	u_{z5}
15	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲
14	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	△
13	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲	▲	▲	△
12	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	△
11	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	△	△	▲	▲	△
10	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	△	△	△	▲	△
9	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	△	△	△	△	△
8	▲	▲	▲	▲	▲	▲	△	△	▲	▲	△	△	△	△	△
7	▲	▲	▲	▲	▲	▲	△	△	▲	△	△	△	△	△	△
6	▲	▲	▲	▲	▲	△	△	△	▲	△	△	△	△	△	△
5	▲	▲	▲	▲	▲	△	△	△	△	△	△	△	△	△	△
4	▲	▲	▲	▲	△	△	△	△	△	△	△	△	△	△	△
3	▲	▲	▲	△	△	△	△	△	△	△	△	△	△	△	△

Lo stesso procedimento è stato poi applicato anche nel caso in cui il file di input derivasse da delle analisi dinamiche, prestando attenzione a opportuni accorgimenti riguardanti l'identificazione delle colonne nel file. La sola differenza dal caso precedente è che stavolta il codice Python è stato impostato per produrre diversi file di output, a seconda dei modi di vibrare della struttura di cui si vuole andare a creare il best theory diagram.

4.2.2 Reti neurali

Lo svolgimento di questa tesi prevede l'implementazione di reti neurali, perciò si rende necessario effettuare in precedenza alcuni test per comprendere meglio il loro funzionamento e come i risultati ottenuti vengano influenzati da diversi parametri.
Per effettuare questi primi confronti è stata utilizzata una rete neurale piuttosto semplice, composta da due hidden layers di tipologia dense e dimensioni rispettivamente 20 e 10 con funzione di attivazione di tipo rectified linear unit (ReLU) ed un layer dense di output di dimensione 1 con funzione di attivazione una sigmoide. La rete è stata impostata per compiere il training su un set contenente il 20% dei dati forniti in input, scelti randomicamente. Il training avviene con l'utilizzo dell'algoritmo Keras Adamax, impostato con un learning rate fisso di 0.001. Un altro parametro che rimane fissato è la separazione tra dati di training e dati di validazione, scelto coincidente con 0.5.
I primi test che sono stati effettutati hanno previsto uno studio sull'influenza dei parametri epochs e batch size. Il primo consiste in quante volte lo step di training viene ripetuto sull'intero set di dati, mentre il secondo indica ogni quanti step di dati viene aggiornato l'errore ed i parametri interni alla rete tramite backpropagation.
La rete è stata quindi allenata con diversi valori di questi due parametri, fornendo in input lo stesso file usato il precedenza per lo sviluppo dei Best Theory Diagrams (BTD). Successivamente è stata utilizzata per fornire una predizione dei risultati di errore attesi in base ai termini attivi della teoria, in modo da poterne effettuare un confronto con i risultati reali.
In figura 4.13 si può notare come l'aumentare del valore di epochs fornisca risultati sempre più simili al risultato reale, ma un risultato quasi accurato lo si ottiene solo per i casi in cui le combinazioni di termini attivi sono molto elevate. Questo comportamento è dovuto al fatto che essendoci molte più combinazioni, ed essendo che la rete sceglie randomicamente quali utilizzare su solamente il 20% dei dati totali, questa è molto allenata nel riprodurre i dati riferiti a quelle combinazioni senza saper però riprodurre le combinazioni viste poche volte, se non addirittura mai. Aumentare il numero di epochs può portare un miglioramento, ma ovviamente il tempo di calcolo aumenta progressivamente. Inoltre, bisogna prestare attenzione che la rete non vada in overfitting, ovvero che vada a memorizzare e quindi riprodurre esattamente i dati che ha già visto, ma che non sappia comportarsi nel caso di nuovi dati forniti. Questo comportamento lo si può intravedere quando è stato utilizzato un valore di epochs pari a 3000, in cui per un certo numero di termini attivi, si ottiene un errore maggiore rispetto alla rete allenata con epochs di 700.

In figura 4.14 vengono riportati i risultati ottenuti con l'utilizzo di differenti valori di batch size. In questo caso, il tempo di calcolo aumenta al diminuire del valore. Diminuendo il valore, il risultato migliora, anche se in entità minore rispetto a quanto visto variando il parametro epochs. Si può inoltre notare che utilizzando batch size=2, è presente un'inversione di tendenza e il risultato peggiora rispetto al caso in cui batch size=4. Questo avviene in quanto avendo un valore di batch size piccolo a fronte di un'elevata quantità di dati di training, l'aggiornamento di pesi e bias diventa

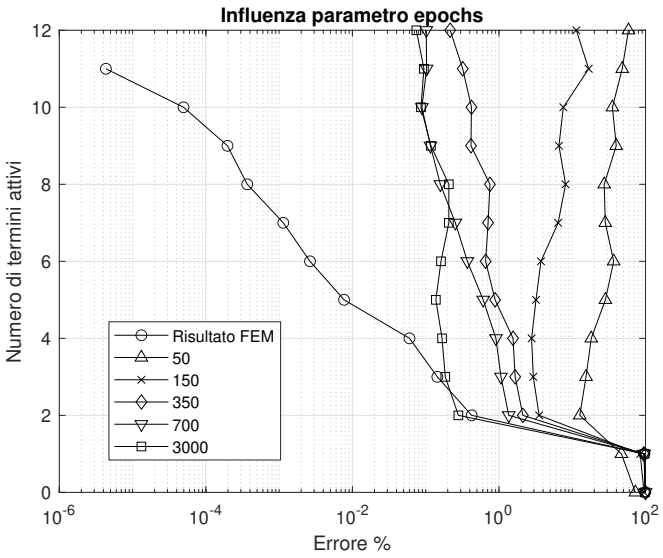


Figura 4.13: Influenza parametro epochs.

altalenante, facendo oscillare la rete tra configurazioni anche molto diverse tra loro. Un batch size più ampio permette di avere errori sui singoli batch più simili tra l'oro, favorendo la convergenza dei parametri della rete in maniera più regolare.

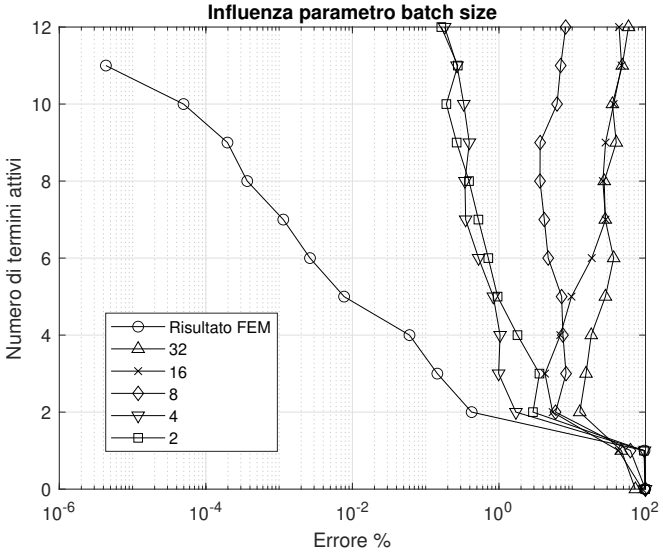


Figura 4.14: Influenza parametro batch size.

Come si può notare, con l'utilizzo di una rete così semplice è difficile riprodurre tutti i risultati. Inoltre, è difficile stabilire quale parametro possa essere più influente rispetto all'altro, tenendo anche in considerazione i tempo di calcolo. Si tratta quindi di effettuare diversi test per trovare la combinazione di parametri che possa fornire il miglior risultato desiderato.

Capitolo 5

Sviluppo rete neurale

5.1 Interfaccia tra rete neurale e software strutturale

In questo capitolo si procede con lo sviluppo della rete neurale che verrà utilizzata in questo lavoro di tesi, ma prima di ciò si vuole andare ad esporre dettagliatamente come avviene l'interfaccia tra il software di calcolo strutturale e la rete neurale artificiale. Questa avviene secondo lo schema riportato in figura 5.1, in cui viene mostrato come software di calcolo strutturale e rete neurale artificiale vengano utilizzati in maniera sequenziale.
Per una facile comprensione, viene esposto separatamente il funzionamento di ciascuna delle due fasi del processo, ovvero la fase I di utilizzo del software e la fase II di utilizzo della rete neurale.

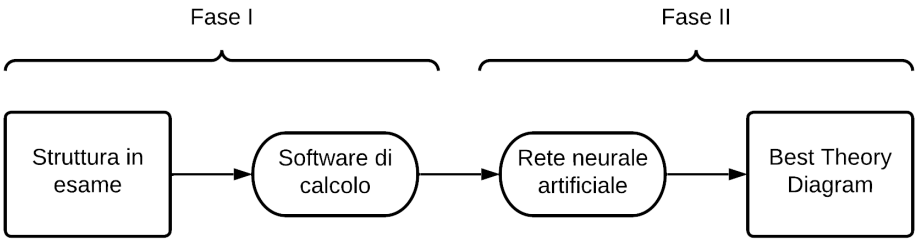


Figura 5.1: Ilustrazione metodologia applicata.

5.1.1 Utilizzo preliminare del software strutturale

Questa è la prima fase fondamentale che viene percorsa per il raggiungimento dell'obiettivo posto per questo lavoro di tesi.
Inizialmente si devono creare i vari file di testo contenenti tutti i dati che rappresentano la struttura da analizzare. Si avranno quindi diversi file, ognuno contenente precise informazioni sul tipo di materiale con le sue proprietà fisiche, la geometria del sistema, le coordinate dei nodi che compongono la mesh, il tipo di elementi fem, le funzioni di espansione con il loro relativo ordine identificate per ogni elemento, i vincoli a cui è

soggetta la struttura ed i carichi agenti su di essa. Oltre a questi, sono necessari altri due file per l'indicazione di che tipo di analisi strutturale eseguire e per la richiesta dei valori da fornire in post-processing.
A seguire, viene avviato il software di calcolo strutturale basato sulla formulazione CUF, che nel lavoro di questa tesi viene utilizzato per analisi che sfruttano il metodo assiomatico-asintotico per la produzione dei risultati. Da questo si ottengono dei file di output contenenti i risultati richiesti, svolti per tutte le combinazioni di termini previste dal metodo AAM, con la metodologia presentata in precedenza nel capitolo teorico. Facendo un riepilogo, si avranno 2^{12} combinazioni nel caso di analisi per l'ottenimento dei Best Theory Diagrams (BTD), in cui vengono considerati i termini attivi delle funzioni di espansione di Taylor fino al quarto grado, mantenendo sempre presenti quelli di ordine zero. Nel caso di ottenimento delle best theory distribution per le analisi global/local invece le combinazioni saranno 2^{16}, poiché la struttura sarà suddivisa con una mesh uniforme 4x4.
Quanto appena esposto è ciò che avviene nella prima fase di questo metodo ed è riportata nello schema di figura 5.2.

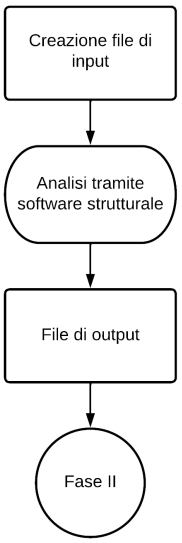


Figura 5.2: Schema di utilizzo del software strutturale.

5.1.2 Utilizzo della rete neurale

La seconda fase del processo prevede l'utilizzo della rete neurale artificiale, servendosi dei dati ottenuti dalla fase precedente.
Vengono dunque importati nell'ambiente di sviluppo Python tutti i dati necessari presenti nel file di output del software di calcolo strutturale. Questi sono la totalità delle combinazioni con tutte le distribuzioni di termini attivi e inattivi, insieme al relativo parametro di controllo su cui si vuole produrre il diagramma BTD, che in questa tesi

coincide con l'errore percentuale relativo sulle prime dieci frequenze naturali per i capitoli 5 e 6, oppure con l'errore percentuale relativo alla misurazione dei failure index nel capitolo 7.
Dopo aver importato tutti i dati necessari, vi è una parte di manipolazione, in cui questi vengono ordinati nei vettori e nelle matrici opportune per il corretto funzionamento. Oltre a questo, viene compiuta una selezione dei dati che verranno utilizzati per la fase di training. Infatti, solo una percentuale dei dati totali forniti dal codice di calcolo andranno ad essere effettivamente impiegati all'interno della rete. Occorre prestare attenzione però al fatto che le varie numerosità di termini attivi hanno differenti insiemi di combinazioni totali, pertanto la scelta dei dati da utilizzare viene effettuata tenendo conto di ciò. In tabella 5.1 viene mostrato un esempio per come avviene la selezione percentuale dei dati a seconda delle numerosità dei gradi di libertà. Questo esempio è nel caso di 2^{16} combinazioni, da cui si vuole utilizzare una percentuale di dati pari al 20% del totale. In base al numero di combinazioni di ogni numerosità di termini (DOF), se ne sceglie una percentuale che sarà maggiore quando sono presenti poche combinazioni e minore quando queste saranno molte. La selezione di questi dati viene comunque eseguita dallo script Phyton ed avviene in maniera casuale, una volta fornita indicazione della percentuale da prelevare in ogni numerosità.

Tabella 5.1: Distribuzione percentuale scelta dati di input.

DOF	Combinazioni	Selezione [%]	Dati Input
0	1	100	1
1	16	100	16
2	120	60	72
3	560	40	224
4	1820	30	546
5	4368	25	1092
6	8008	20	1602
7	11440	17	1945
8	12870	17	2188
9	11440	17	1945
10	8008	20	1602
11	4368	25	1092
12	1820	30	546
13	560	40	224
14	120	60	72
15	16	100	16
16	1	100	1
	Totale	Selezione [%]	Totale dati
	65536	20	13184

Dopo la manipolazione dei dati e dopo aver fornito indicazione alla rete su quali siano quelli di input e quali quelli di ouput, questi vengono divisi a metà per la fase di training della rete neurale, andando ad essere utilizzati la prima metà per il training vero e proprio e l'altra metà per la validazione dei risultati. Questa fase avviene

seguendo l'architettura impostata per la rete, con i vari parametri definiti per il processo. Una volta conclusa, viene conservata in memoria solamente la combinazione di pesi e funzioni interni alla rete che hanno portato ai risultati migliori nella validazione. A questo punto la rete neurale è pronta per essere utilizzata per il suo scopo. Vengono forniti in input solamente le informazioni riguardanti le combinazioni di termini attivi, insieme a, nel caso dei failure index, le coordinate dei punti, e la rete effettua la prediction sui valori dei paramentri di controllo, in base a come è stata allenata in precedenza. I risultati vengono poi riportati in file di testo, creando il BTD finale e confrontando quello ottenuto tramite prediction effettuato dalla rete neurale artificiale, con quello prodotto dall'analisi fem effettuata dal codice di calcolo.
In questa maniera termina la seconda ed ultima fase del metodo e ne viene proposto uno schema riassuntivo in figura 5.3.

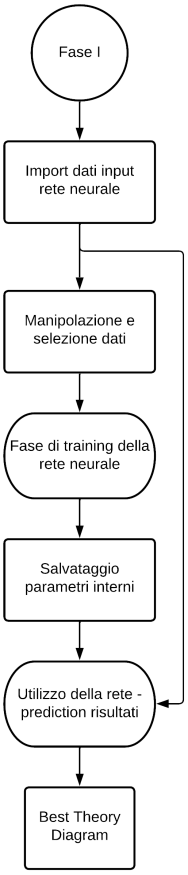


Figura 5.3: Schema di utilizzo della rete neurale.

5.2 Rete neurale multi-output

In questa sezione si affronta ora la problematica di sviluppare una rete neurale che possa fornire un risultato multidimensionale accurato e non più uno scalare. Per fare

ciò, si cerca di costruire una rete con cui riprodurre dei best theory diagrams riferiti ad un'analisi dinamica di una piastra semplicemente appoggiata, laminazione 0/90, a/h=100, R/a=20, in cui si ricercano gli errori percentuali minori sul risultato riferito alle prime dieci frequenze proprie della struttura.
Il procedimento adottato è stato quello di impostare un'architettura di rete neurale ed andare a variare alcuni suoi parametri con l'obiettivo di migliorare l'accuratezza del risultato.
Sono stati creati tre script python:

- Il primo script, denominato main, consiste nella struttura fondamentale della rete neurale, in cui vengono settati input, output ed i vari parametri di training e validazione. Al suo interno viene anche effettuata la procedura di training della rete ed il suo rispettivo salvataggio in modo da renderla facilmente richiamabile.

- Un secondo script si occupa di richiamare la rete allenata in precedenza ed utilizzarla per ottenere la prediction sugli errori percentuali delle prime dieci frequenze, fornendo come input un file derivante da un'analisi dinamica. Vengno poi prodotti due file di output. Il primo contenente i valori della prediction della rete accoppiati con i corrispondenti valori dell'analisi dinamica, mentre il secondo contenente i soli valori della prediction della rete, in cui ad ogni colonna corrisponde una frequenza in ordine crescente.

- Il terzo script si occupa dell'analisi dei dati ottenuti con il precedente, andando a sviluppare il BTD sia dell'analisi dinamica, sia delle prediction della rete neurale. I risultati vengono poi salvati e presentati in ordine di frequenza, accoppiati a due a due tra analisi dinamica e prediction per ogni numerosità di termini attivi.

5.2.1 Architettura rete

Nel descrivere l'architettura utilizzata per la rete è necessario inziare con la descrizione di quali siano i dati forniti per l'input. Viene richiamato all'interno dello script il file ottenuto come output dall'analisi dinamica sulla struttura di tipo piastra. Il file presenta ventiquattro colonne di dati in cui la prima colonna rappresenta un indice progressivo, la seconda riporta il numero di termini attivi nella teoria utilizzata dal software di calcolo, le colonne dalla terza alla quattordicesima indicano quali siano i termini attivi e quali no della teoria, le ultime dieci colonne riportano l'errore percentuale sul risultato calcolato con ogni teoria sulle prime dieci frequenze del sistema. I primi tre termini della teoria sono stati considerati sempre attivi, perciò quelli riportati sono i successivi. Il totale dei casi valutati, coincidenti con il numero di righe di dati del file, corrisponde a 4096, ottenibile effettuando la combinazione dei termini considerati ($2^{12} = 4096$).
Da questi dati vengono create due differenti matrici, una contenente tutte le colonne dei termini attivi ed una contenente tutte le colonne degli errori. Successivamente vi è una parte in cui questi dati vengono manipolati, in modo da poter creare i set di input e di validazione per il training della rete. Quello che è stato fatto è stato andare a

selezionare arbitrariamente diverse percentuali di set di dati per ciascuna numerosità di termini attivi della teoria. Questo procedimento è stato effettuato ragionando sul fatto che ci si trova a lavorare con una scarsità di dati e che alcune numerosità hanno una quantità molto maggiore di combinazioni rispetto ad altre. In questa maniera si riesce parzialmente ad equilibrare le quantità di dati e le tipologie di combinazioni su cui la rete neurale andrà ad effettuare il training. Si fa notare che la suddivisione percentuale delle numerosità è stata scelta arbitrariamente, ma la scelta dei termini viene effettuata in maniera randomica ogni volta che lo script viene eseguito. Come ultima procedura di manipolazione dei dati, viene creata la forma dell'input corrispondente ad una sequenza di tensori di dimensione $(4, 3, 1)$ per la totalità dei dati. Gli output vengono invece normalizzati dividendoli per 100, in maniera da avere solo valori che possano variare tra lo zero e l'unità.
A questo punto viene definita, in ordine progressivo, la sequenza di layer che compongono la rete neurale, di cui se ne può vedere la struttura in tabella 5.2.

Tabella 5.2: Architettura rete neurale - output multidimensionale.

Tipologia	Dimensione	Nucleo	Passo	Bias	Funzione di attivazione
Convolutional 2D	128	3x3	1	1	ReLU
Convolutional 2D	128	3x3	1	1	ReLU
Convolutional 2D	128	3x3	1	1	ReLU
Flatten	-	-	-	-	-
Dense	128	-	-	-	ReLU
Dense	128	-	-	-	ReLU
Dense	10	-	-	-	Sigmoid

Viene utilizzata la funzione di ottimizzazione dell'errore tramite backpropagation *Adamax*, di cui se ne può trovare una descrizione accurata in [29], accoppiata ad una valutazione dell'errore tramite la funzione di calcolo dell'errore quadratico medio:

$$loss = \sqrt{y_{true} - y_{predict}} \tag{5.1}$$

I parametri interni della rete vengono aggiornati durante il training e, di questi, ne viene salvata in memoria solamente la combinazione che fornisce il risultato migliore. Il training è stato effettuato diverse volte, andando a modificarne arbitrariamente i parametri. Le indicazioni dei valori utilizzati ad ogni training sono riportate nella sezione successiva.

5.2.2 Variazione dei parametri

La variazione dei parametri con cui viene effettuato il training è uno step fondamentale per la costruzione della rete neurale, in quanto si vuole andare a costruire un modello efficace che sia accurato e che possa fornire i risultati in un tempo ragionevolmente accettabile. Tutti i test sono stati svolti sullo stesso hardware, in modo da poter confrontare i tempi di calcolo necessari.

Ogni test è stato effettuato andando a variare generalmente uno solo dei vari parametri, in modo tale da poter comprendere al meglio quali fossero i cambiamenti in grado di apportare una modifica positiva alla rete neurale, sia in termini di accuratezza che di vantaggio nei tempi di calcolo. Nel caso in cui la variazione di un parametro fosse in grado di portare un beneficio, i test successivi sono stati svolti concentrandosi su quella variazione per capire fino a che punto fosse in grado di produrre un miglioramento notevole all'interno della rete, prima di procedere con il considerare la combinazione della variazione di più parametri.
Sono stati svolti diverse decine di test fino al raggiungimento di alcune caratteristiche costanti che permettono alla rete neurale artificiale di fornire buoni risultati. Vengono dunque riportati i valori stabiliti per i parametri fondamentali nella tabella 5.3.

Tabella 5.3: Parametri finali rete neurale artificiale multi-output.

Parametro	Valore
Learning rate	0.001
Epochs	500
Batch size	8

Oltre a questi parametri, sono stati fatti variare anche la dimensione del nucleo dei layer convolutional, la dimensione dei layer stessi, l'utilizzo dei parametri di bias e le funzioni di attivazione, ma nessuno di questi tentativi ha portato ad un miglioramento dei risultati predetti dalla rete. Inoltre, era stato riscontrato un certo pattern nella predizione dei dati di output, per cui alcuni risultavano essere più accurati di altri senza un apparente motivo. Questo è stato risolto andando ad introdurre un rimescolamento dell'ordine in cui i dati venivano forniti al training, portando ad esiti positivi e ad un miglioramento complessivo del funzionamento della rete neurale. Un successivo step è stato quello di ridurre i dati forniti in input alla rete neurale, passando da un 40% dei dati, sul totale di quelli ottenuti dall'analisi dinamica, come spiegato in precedenza, ad un 20% ed andando a suddividerli a metà tra il training e la validazione. Soffermandosi poi sulle prediction dei casi in cui l'errore percentuale sulle frequenze risultava essere molto piccolo, con diversi ordini di grandezza inferiori allo zero, è stato valutato l'effetto di utilizzare differenti loss function per la valutazione dell'errore, riuscendo così a stabilire quale fosse la più opportuna da utilizzare sempre con lo scopo di migliorare l'accuratezza della rete neurale. Sono state così valutate le funzioni di errore logaritmico quadratico medio ed errore assoluto percentuale medio, di cui di quest'ultima ne viene riportata la relazione, essendo quella che implementandola ha portato esiti migliori:

$$loss = 100 \times |\frac{y_{true} - y_{predict}}{y_{true}}| \tag{5.2}$$

A seguito dei vari test è stato possibile individuare quattro casistiche di interesse, che vengono riportate a seguire indicandone le differenze:

- **Caso A:** 40% dei dati totali, corrispondenti a circa 1600 dati, utilizzati 800 per la fase di training ed 800 per la fase di validazione. Calcolo dell'errore tramite funzione *MeanSquaredError* (5.1).
- **Caso B:** 20% dei dati totali, corrispondenti a circa 800 dati, utilizzati 400 per la fase di training e 400 per la fase di validazione. Calcolo dell'errore tramite funzione *MeanSquaredError* (5.1).
- **Caso C:** 40% dei dati totali, corrispondenti a circa 1600 dati, utilizzati 800 per la fase di training ed 800 per la fase di validazione. Calcolo dell'errore tramite funzione *MeanAbsolutePercentageError* (5.2).
- **Caso D:** 20% dei dati totali, corrispondenti a circa 800 dati, utilizzati 400 per la fase di training e 400 per la fase di validazione. Calcolo dell'errore tramite funzione *MeanAbsolutePercentageError* (5.2).

5.2.3 Risultati

I risultati ottenuti dai vari test presentano alcune peculiarità comuni. La rete neurale non riesce a predire bene gli errori percentuali delle numerosità più alte, corrispondenti ad errori con ordini di grandezza molto bassi quando si utilizza come funzione di perdita l'errore quadratico medio. Inoltre, non sempre l'errore minino associato ad una numerosità il cui risultato è molto simile corrisponde alla stessa distribuzione di termini attivi che si ottiene dal BTD estrapolato dall'analisi dinamica. Si osserva anche che la nona frequenza del sistema sia quella che viene meglio riprodotta dalla rete.
Nello svolgere i vari test, si era provato ad aumentare il parametro epochs ed a dimunire il parametro di batch size con finalità di migliorare l'accuratezza dei risultati. Quello che si è ottenuto però è stata una variazione non significativa, a scapito di un aumento progressivo del tempo di calcolo, passando da pochi minuti fino ad oltre mezz'ora. Per questo motivo, dopo i primi test si è tornati ai valori iniziali di questi due parametri. Ad un certo punto si è andati a manipolare, oltre al learning rate, anche i dati forniti alla rete per la fase di training e si sono ottenuti i risultati migliori, in particolare sotto l'aspetto della distribuzione dei termini attivi del BTD ottenuto dalla rete con quello ottenuto dall'analisi dinamica. Nei successivi test si è voluto continuare con la manipolazione dei dati.
Nelle tabelle a seguire, vengono riportati gli errori percentuali riferiti ai BTD ottenuti con l'analisi dinamica e nei casi A, B, C, D per ognuna delle prime dieci frequenze del sistema, per poterne effettuare un confronto. Si può notare come la diminuzione dei dati a disposizione dalla rete per il processo di training vada a peggiorare lievemente le prediction, pertanto sarebbe più opportuno poter aumentare la quantità di dati a disposizione, prestando attenzione che la rete non vada in over-fitting.

Tabella 5.4: Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza I.

DOF	Err% Dyn	Caso **A**	Caso **B**	Caso **C**	Caso **D**
3	99.797	99.814	99.935	99.837	99.809
4	99.594	99.621	99.467	99.582	99.277
5	1.713	1.321	1.582	1.718	2.105
6	0.047	0.121	0.182	0.047	0.060
7	0.016	0.080	0.104	0.016	0.029
8	8.365×10^{-3}	4.777×10^{-2}	8.125×10^{-2}	9.441×10^{-3}	1.263×10^{-2}
9	5.030×10^{-3}	3.978×10^{-2}	5.671×10^{-2}	5.466×10^{-3}	8.217×10^{-3}
10	2.379×10^{-3}	2.781×10^{-2}	5.233×10^{-2}	2.836×10^{-3}	3.400×10^{-3}
11	7.364×10^{-4}	3.015×10^{-2}	4.225×10^{-2}	1.050×10^{-3}	1.197×10^{-3}
12	8.447×10^{-5}	2.214×10^{-2}	3.575×10^{-2}	8.448×10^{-5}	1.525×10^{-4}
13	6.611×10^{-5}	1.920×10^{-2}	3.304×10^{-2}	6.611×10^{-5}	3.437×10^{-5}
14	5.483×10^{-7}	1.838×10^{-2}	2.960×10^{-2}	3.007×10^{-6}	4.617×10^{-6}
15	8.380×10^{-11}	1.801×10^{-2}	4.731×10^{-2}	3.675×10^{-5}	1.835×10^{-5}

Tabella 5.5: Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza II.

DOF	Err% Dyn	Caso **A**	Caso **B**	Caso **C**	Caso **D**
3	99.030	98.957	98.828	99.027	98.858
4	91.099	91.381	90.422	91.004	90.996
5	2.332	2.526	2.593	2.238	2.682
6	1.785	1.567	1.106	0.662	0.869
7	0.385	0.767	0.731	0.287	0.328
8	8.715×10^{-2}	4.904×10^{-1}	5.449×10^{-1}	1.094×10^{-1}	1.454×10^{-1}
9	4.126×10^{-2}	3.295×10^{-1}	4.266×10^{-1}	5.317×10^{-2}	3.958×10^{-2}
10	4.102×10^{-2}	2.805×10^{-1}	4.211×10^{-1}	7.452×10^{-3}	1.245×10^{-2}
11	2.172×10^{-3}	2.444×10^{-1}	3.600×10^{-1}	1.673×10^{-3}	4.550×10^{-3}
12	2.600×10^{-4}	1.924×10^{-1}	3.175×10^{-1}	2.600×10^{-4}	1.042×10^{-3}
13	2.157×10^{-5}	1.751×10^{-1}	3.578×10^{-1}	2.157×10^{-5}	3.312×10^{-4}
14	2.464×10^{-4}	2.260×10^{-1}	3.519×10^{-1}	4.204×10^{-5}	2.436×10^{-4}
15	3.104×10^{-11}	2.772×10^{-1}	4.964×10^{-1}	6.256×10^{-4}	3.732×10^{-4}

Tabella 5.6: Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza III.

DOF	Err% Dyn	Caso **A**	Caso **B**	Caso **C**	Caso **D**
3	99.021	99.052	99.048	99.005	98.917
4	94.313	94.590	93.767	94.291	94.161
5	2.588	1.512	2.022	2.629	2.728
6	0.015	0.391	0.710	0.015	0.544
7	0.015	0.256	0.502	0.015	0.235
8	1.558×10^{-2}	2.090×10^{-1}	3.866×10^{-1}	3.991×10^{-2}	1.013×10^{-1}
9	1.587×10^{-2}	1.825×10^{-1}	3.115×10^{-1}	4.580×10^{-2}	4.233×10^{-2}
10	1.588×10^{-2}	1.924×10^{-1}	3.628×10^{-1}	9.994×10^{-3}	8.424×10^{-3}
11	7.744×10^{-3}	1.759×10^{-1}	3.203×10^{-1}	5.451×10^{-3}	6.537×10^{-3}
12	3.099×10^{-3}	1.707×10^{-1}	2.932×10^{-1}	2.978×10^{-3}	4.015×10^{-3}
13	1.403×10^{-3}	1.848×10^{-1}	2.777×10^{-1}	1.419×10^{-3}	2.151×10^{-3}
14	1.271×10^{-3}	2.119×10^{-1}	2.774×10^{-1}	1.265×10^{-3}	1.273×10^{-3}
15	8.491×10^{-12}	2.826×10^{-1}	3.567×10^{-1}	7.172×10^{-3}	1.917×10^{-3}

Tabella 5.7: Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza IV.

DOF	Err% Dyn	Caso **A**	Caso **B**	Caso **C**	Caso **D**
3	98.782	98.821	98.670	98.781	98.324
4	94.560	94.784	94.227	94.522	94.624
5	2.561	2.068	2.240	2.560	2.875
6	0.136	0.203	0.329	0.144	0.195
7	0.088	0.126	0.195	0.095	0.094
8	4.990×10^{-2}	7.580×10^{-2}	1.543×10^{-1}	4.860×10^{-2}	5.572×10^{-2}
9	3.242×10^{-2}	7.036×10^{-2}	1.094×10^{-1}	3.282×10^{-2}	3.381×10^{-2}
10	2.137×10^{-2}	5.117×10^{-2}	8.507×10^{-2}	2.137×10^{-2}	2.303×10^{-2}
11	1.181×10^{-2}	4.990×10^{-2}	6.835×10^{-2}	1.159×10^{-2}	9.873×10^{-3}
12	4.551×10^{-3}	4.360×10^{-2}	5.833×10^{-2}	6.185×10^{-3}	6.939×10^{-4}
13	1.837×10^{-3}	3.949×10^{-2}	5.345×10^{-2}	1.639×10^{-3}	6.626×10^{-5}
14	5.869×10^{-6}	3.573×10^{-2}	4.786×10^{-2}	6.008×10^{-6}	5.843×10^{-6}
15	2.814×10^{-12}	3.651×10^{-2}	7.414×10^{-2}	1.038×10^{-4}	2.479×10^{-5}

Tabella 5.8: Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza V.

DOF	Err% Dyn	Caso **A**	Caso **B**	Caso **C**	Caso **D**
3	97.281	97.352	97.021	97.421	96.691
4	92.034	92.196	91.361	91.972	92.086
5	2.618	2.361	2.918	2.680	3.188
6	1.091	1.250	1.339	1.310	1.297
7	0.775	0.741	0.836	0.805	0.644
8	2.786×10^{-1}	4.857×10^{-1}	6.549×10^{-1}	3.140×10^{-1}	2.869×10^{-1}
9	1.217×10^{-1}	3.368×10^{-1}	4.778×10^{-1}	1.028×10^{-1}	5.396×10^{-2}
10	4.289×10^{-3}	2.554×10^{-1}	4.339×10^{-1}	6.629×10^{-3}	1.573×10^{-2}
11	2.370×10^{-3}	2.097×10^{-1}	3.649×10^{-1}	2.372×10^{-3}	6.646×10^{-3}
12	1.967×10^{-3}	1.753×10^{-1}	3.251×10^{-1}	7.988×10^{-4}	1.678×10^{-3}
13	2.257×10^{-4}	1.640×10^{-1}	3.165×10^{-1}	2.036×10^{-4}	5.401×10^{-4}
14	2.171×10^{-4}	1.885×10^{-1}	3.021×10^{-1}	2.172×10^{-4}	2.202×10^{-4}
15	6.253×10^{-12}	2.513×10^{-1}	3.590×10^{-1}	1.639×10^{-3}	3.437×10^{-4}

Tabella 5.9: Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza VI.

DOF	Err% Dyn	Caso **A**	Caso **B**	Caso **C**	Caso **D**
3	97.263	97.407	97.210	97.242	96.632
4	92.063	92.271	91.460	92.031	91.874
5	2.792	2.195	2.406	2.792	4.003
6	1.266	0.831	0.944	0.377	1.651
7	0.029	0.403	0.544	0.187	0.824
8	2.609×10^{-2}	3.139×10^{-1}	3.879×10^{-1}	1.460×10^{-1}	3.083×10^{-1}
9	1.731×10^{-2}	2.653×10^{-1}	2.745×10^{-1}	3.570×10^{-2}	9.529×10^{-2}
10	1.760×10^{-2}	2.153×10^{-1}	2.916×10^{-1}	5.245×10^{-3}	3.646×10^{-2}
11	5.781×10^{-3}	1.708×10^{-1}	2.751×10^{-1}	1.793×10^{-3}	7.221×10^{-3}
12	3.377×10^{-3}	1.399×10^{-1}	2.385×10^{-1}	5.671×10^{-4}	2.489×10^{-3}
13	2.622×10^{-3}	1.534×10^{-1}	2.044×10^{-1}	3.675×10^{-4}	1.440×10^{-3}
14	1.840×10^{-3}	1.607×10^{-1}	1.926×10^{-1}	8.642×10^{-4}	9.388×10^{-4}
15	2.168×10^{-12}	2.236×10^{-1}	2.411×10^{-1}	1.832×10^{-3}	1.738×10^{-3}

Tabella 5.10: Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza VII.

DOF	Err% Dyn	Caso **A**	Caso **B**	Caso **C**	Caso **D**
3	97.527	97.465	97.288	97.519	97.244
4	90.730	90.969	90.052	90.683	90.666
5	2.680	2.378	2.699	2.813	3.152
6	0.602	0.739	0.711	0.764	0.716
7	0.513	0.509	0.476	0.439	0.479
8	3.156×10^{-1}	3.798×10^{-1}	3.578×10^{-1}	2.132×10^{-1}	2.692×10^{-1}
9	1.511×10^{-1}	2.577×10^{-1}	2.594×10^{-1}	1.138×10^{-1}	1.234×10^{-1}
10	2.424×10^{-2}	1.879×10^{-1}	2.191×10^{-1}	1.243×10^{-2}	4.555×10^{-2}
11	2.372×10^{-2}	1.674×10^{-1}	1.733×10^{-1}	4.621×10^{-3}	1.907×10^{-2}
12	1.766×10^{-3}	1.430×10^{-1}	1.334×10^{-1}	1.766×10^{-3}	2.817×10^{-3}
13	6.004×10^{-4}	1.242×10^{-1}	1.369×10^{-1}	6.145×10^{-4}	4.734×10^{-4}
14	1.877×10^{-4}	1.246×10^{-1}	1.201×10^{-1}	1.877×10^{-4}	1.315×10^{-4}
15	1.379×10^{-1}	1.276×10^{-1}	1.551×10^{-1}	3.916×10^{-4}	8.798×10^{-5}

Tabella 5.11: Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza VIII.

DOF	Err% Dyn	Caso **A**	Caso **B**	Caso **C**	Caso **D**
3	97.810	97.574	97.354	97.385	96.874
4	92.047	92.358	91.567	92.048	91.293
5	2.720	2.192	2.366	2.643	3.353
6	0.648	0.578	0.558	0.513	0.886
7	0.228	0.390	0.366	0.243	0.501
8	6.546×10^{-2}	2.486×10^{-1}	2.684×10^{-1}	8.856×10^{-2}	1.209×10^{-1}
9	3.248×10^{-2}	1.695×10^{-1}	1.910×10^{-1}	1.386×10^{-2}	2.034×10^{-3}
10	5.471×10^{-4}	1.160×10^{-1}	1.698×10^{-1}	4.094×10^{-3}	2.214×10^{-4}
11	5.854×10^{-4}	9.056×10^{-2}	1.370×10^{-1}	1.280×10^{-3}	3.214×10^{-5}
12	3.010×10^{-3}	7.192×10^{-2}	1.134×10^{-1}	7.709×10^{-4}	6.214×10^{-6}
13	3.182×10^{-4}	6.086×10^{-2}	9.497×10^{-2}	1.203×10^{-4}	6.612×10^{-7}
14	1.939×10^{-5}	6.394×10^{-2}	8.449×10^{-2}	7.640×10^{-6}	4.524×10^{-8}
15	1.381×10^{-1}	6.811×10^{-2}	1.153×10^{-1}	4.696×10^{-6}	4.249×10^{-8}

Tabella 5.12: Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza IX.

DOF	Err% Dyn	Caso **A**	Caso **B**	Caso **C**	Caso **D**
3	95.035	95.119	95.175	95.160	95.007
4	85.813	86.114	85.344	85.686	85.811
5	2.726	2.304	2.540	2.776	2.846
6	0.374	0.412	0.502	0.392	0.494
7	0.250	0.279	0.319	0.272	0.277
8	1.366×10^{-1}	1.770×10^{-1}	2.473×10^{-1}	1.381×10^{-1}	1.582×10^{-1}
9	9.250×10^{-2}	1.441×10^{-1}	1.745×10^{-1}	9.480×10^{-2}	9.185×10^{-2}
10	5.476×10^{-2}	1.051×10^{-1}	1.367×10^{-1}	5.480×10^{-2}	6.268×10^{-2}
11	2.946×10^{-2}	9.113×10^{-2}	1.110×10^{-1}	3.053×10^{-2}	2.258×10^{-2}
12	6.413×10^{-3}	7.6264×10^{-2}	8.526×10^{-2}	1.093×10^{-2}	1.457×10^{-3}
13	3.136×10^{-3}	6.692×10^{-2}	9.050×10^{-2}	3.277×10^{-3}	1.684×10^{-4}
14	1.762×10^{-5}	6.123×10^{-2}	7.908×10^{-2}	1.759×10^{-5}	1.853×10^{-5}
15	2.488×10^{-13}	6.041×10^{-2}	1.120×10^{-1}	2.494×10^{-4}	3.998×10^{-5}

Tabella 5.13: Semplice appoggio, 0/90, a/h=100, R/a=20 - Frequenza X.

DOF	Err% Dyn	Caso **A**	Caso **B**	Caso **C**	Caso **D**
3	91.366	92.220	92.276	92.348	91.787
4	84.653	85.023	83.881	84.596	84.631
5	2.932	2.950	3.211	2.932	3.713
6	1.558	1.636	1.731	1.539	1.889
7	0.924	1.108	1.153	0.922	1.227
8	5.928×10^{-1}	7.641×10^{-1}	8.424×10^{-1}	5.875×10^{-1}	5.612×10^{-1}
9	2.503×10^{-1}	3.759×10^{-1}	5.319×10^{-1}	1.849×10^{-1}	8.919×10^{-2}
10	3.453×10^{-3}	2.588×10^{-1}	4.418×10^{-1}	4.434×10^{-3}	2.017×10^{-2}
11	2.892×10^{-3}	1.833×10^{-1}	3.558×10^{-1}	1.239×10^{-3}	5.637×10^{-3}
12	7.649×10^{-4}	1.588×10^{-1}	2.982×10^{-1}	7.791×10^{-4}	1.318×10^{-3}
13	4.569×10^{-4}	1.379×10^{-1}	2.620×10^{-1}	4.569×10^{-4}	6.107×10^{-4}
14	3.044×10^{-4}	1.291×10^{-1}	2.498×10^{-1}	3.045×10^{-4}	3.217×10^{-4}
15	6.923×10^{-13}	1.601×10^{-1}	3.107×10^{-1}	1.032×10^{-3}	3.597×10^{-4}

Tabella 5.14: Semplice appoggio, 0/90, a/h=100, R/a=20 - Media Frequenze.

DOF	Err% Dyn	Caso **A**	Caso **B**	Caso **C**	Caso **D**
3	97.291	97.378	97.281	97.373	97.014
4	91.691	91.931	91.155	91.642	91.542
5	2.566	2.181	2.458	2.578	3.065
6	0.752	0.773	0.811	0.576	0.860
7	0.322	0.466	0.523	0.328	0.464
8	1.576×10^{-01}	3.192×10^{-01}	3.926×10^{-01}	1.695×10^{-01}	2.020×10^{-01}
9	7.600×10^{-02}	2.171×10^{-01}	2.813×10^{-01}	6.831×10^{-02}	5.797×10^{-02}
10	1.855×10^{-02}	1.690×10^{-01}	2.614×10^{-01}	1.293×10^{-02}	2.281×10^{-02}
11	8.727×10^{-03}	1.413×10^{-01}	2.208×10^{-01}	6.160×10^{-03}	8.334×10^{-03}
12	2.529×10^{-03}	1.194×10^{-01}	1.899×10^{-01}	2.512×10^{-03}	1.567×10^{-03}
13	1.069×10^{-03}	1.126×10^{-01}	1.827×10^{-01}	8.185×10^{-04}	5.816×10^{-04}
14	4.110×10^{-04}	1.220×10^{-01}	1.735×10^{-01}	2.915×10^{-04}	3.158×10^{-04}
15	2.760×10^{-02}	1.505×10^{-01}	2.268×10^{-01}	1.309×10^{-03}	4.903×10^{-04}

Osservando i valori riportati nelle varie tabelle di ogni frequenza, si nota come effettivamente i casi C e D riescano a produrre un'accuratezza generalmente migliore rispetto ai casi A e B, riportati nelle tabelle in quanto sono stati uno step fondamentale nel raggiungimento di questi risultati. Ancora più notevole risulta essere la differenza quando i gradi di libertà sono elevati. Questo avviene perché appunto la scelta di quella specifica funzione per il calcolo dell'errore risulta più efficiente rispetto all'altra nell'operare con numeri molto piccoli e vicini allo zero. L'utilizzo del 40% dei dati aiuta la rete neurale a saper predirre meglio i risultati attesi, anche se questo non sempre succede, tanto che talvolta utilizzando il 20% dei dati, sono stati ottenuti valori più simili a quelli della dinamica. La variabilità di questa accuratezza è dovuta al fatto che i dati di input vengono scelti ogni volta casualmente, pertanto risulta essere possibile che alla rete neurale arrivi in input una combinazione più accurata nel caso in cui si prelevino meno dati, e non nel caso di una maggioranza, dovuto proprio alla randomicità con cui è necessario operare.
Ovviamente l'utilizzo di un set minore di dati di input, permette alla rete neurale di operare con un costo computazionale minore, diventando così più rapida nel suo funzionamento. Per questo motivo, e vista l'ottima accuratezza ottenuta, si è deciso di considerare il caso **D** come architettura definitiva della rete neurale artificiale convoluzionale che sarà utilizzata da qui in poi nello svolgimento di questa tesi.
Viene ora riportata la distribuzione di termini attivi per le tre casistiche a cinque, otto e dodici gradi di libertà per ognuna delle dieci frequenze predette dalla rete neurale, proponendone un confronto con la media delle dieci frequenze riportata dallo studio eseguito in [19]. La tipologia di schema utilizzato nella rappresentazione segue la distribuzione di termini secondo l'esempio della tabella 5.15. Inoltre, viene introdotto l'utilizzo del colore rosso per distinguere quei termini attivi della media e di ogni frequenza, la cui attivazione non avviene nella totalità delle distribuzioni. Nella tabella

5.16 viene riportata una legenda di riferimento per i colori utilizzati.

Tabella 5.15: Disposizione dei termini.

N = 0	N = 1	N = 2	N = 3	N = 4
u_{x_1}	u_{x_2}	u_{x_3}	u_{x_4}	u_{x_5}
u_{y_1}	u_{y_2}	u_{y_3}	u_{y_4}	u_{y_5}
u_{z_1}	u_{z_2}	u_{z_3}	u_{z_4}	u_{z_5}

Tabella 5.16: Legenda colori.

▲	Termine attivo, sempre presente
▲	Termine attivo, non sempre presente
△	Termine non attivo

Tabella 5.17: BTD complessivo 5 DOF - Caso **D**.

Media [19]					Frequenza I					Frequenza II					Frequenza III				
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△
Frequenza IV					Frequenza V					Frequenza VI					Frequenza VII				
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△
Frequenza VIII					Frequenza IX					Frequenza X									
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△					
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△					
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△					

Tabella 5.18: BTD complessivo 8 DOF - Caso **D**.

Media [19]					Frequenza I					Frequenza II					Frequenza III				
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	▲	△	△	▲	▲	△	△	△
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	▲
▲	▲	▲	△	△	▲	▲	▲	△	△	▲	▲	▲	△	△	▲	▲	▲	△	△

Frequenza IV					Frequenza V					Frequenza VI					Frequenza VII				
▲	▲	△	▲	△	▲	▲	▲	△	△	▲	▲	△	△	△	▲	▲	△	▲	△
▲	▲	△	▲	△	▲	▲	△	△	△	▲	▲	▲	△	△	▲	▲	△	△	△
▲	△	▲	△	△	▲	▲	▲	△	△	▲	▲	▲	△	△	▲	△	▲	▲	△

Frequenza VIII					Frequenza IX					Frequenza X				
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	▲	△	△
▲	▲	△	△	△	▲	▲	△	▲	△	▲	▲	△	△	△
▲	▲	▲	△	△	▲	△	▲	△	△	▲	▲	▲	△	△

Tabella 5.19: BTD complessivo 12 DOF - Caso **D**.

Media [19]					Frequenza I					Frequenza II					Frequenza III				
▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲	▲	▲	▲	▲	△
▲	▲	△	▲	△	▲	▲	▲	▲	▲	▲	▲	△	△	▲	▲	▲	▲	▲	▲
▲	▲	▲	△	▲	▲	▲	▲	△	△	▲	▲	▲	△	▲	▲	▲	▲	△	△

Frequenza IV					Frequenza V					Frequenza VI					Frequenza VII				
▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲
▲	▲	▲	▲	△	▲	▲	△	△	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	△
▲	▲	▲	△	△	▲	▲	▲	△	▲	▲	▲	▲	△	△	▲	▲	▲	△	△

Frequenza VIII					Frequenza IX					Frequenza X				
▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲
▲	▲	△	▲	▲	▲	▲	▲	▲	△	▲	▲	△	△	▲
▲	▲	▲	△	△	▲	▲	▲	△	△	▲	▲	▲	△	▲

Le varie distribuzioni di termini attivi ottenute utilizzando la rete neurale risultano essere molto simili a quella della media delle prime dieci frequenze naturali del sistema calcolata dai software di calcolo e già verificata in [19]. Nel caso di cinque DOF la distribuzione è identica per ogni frequenza, mentre nel caso ad otto DOF vi è generalmente un solo termine che si trova attivato in posizione differente. Nell'ultimo confronto riportato, quello a dodici DOF sono presenti frequenze in cui un solo termine risulta spostato, mentre in alcune sono due i termini ad essere non corretti. Complessivamente si tratta di buoni risultati trattandosi sempre di termini di ordine elevato e che quindi hanno un'influenza relativamente bassa sull'accuratezza della teoria. Inoltre, il confronto viene effettuato per ogni frequenza con la media delle prime

dieci frequenze, pertanto alcune incongruenze derivano anche da questa scelta.

5.3 Analisi vibrazioni libere

A questo punto che l'architettura della rete neurale artificale è stata definita e ne è stata verificata l'affidabilità, se ne vuole sperimentare ulteriormente le potenzialità andandola ad applicare per la valutazione dei BTD delle prime frequenze naturali di elementi di tipo guscio, in cui viene fatta variare la curvatura o lo spessore, oltre alle condizioni di vincolo.
Le caratteristiche della struttura in esame corrispondono a quanto analizzato in [19]. Le dimensioni sono $a = b$, con $R_\alpha = R_\beta = R$. Le proprietà del materiale sono $E_1/E_2 = 25,\ G_{12}/E_2 = G_{13}/E_2 = 0.5,\ G_{13}/E_2 = 0.2,\ \nu = 0.25$.
La procedura seguita consiste in una prima analisi dinamica della struttura utilizzando espansioni di Taylor N=4, in modo da poter verificare l'affidabilità del codice di calcolo strutturale confrontando la prima frequenza naturale di ogni modello con i dati già verificati in [19]. Conseguentemente ad esito positivo di questa prima operazione, viene effettuata un'analisi tramite il codice di calcolo che sfrutta l'approccio AAM per la determinazione del BTD della media delle prime dieci frequenze naturali, che verrà anch'esso confrontato con i risultati già validati. La fase di accertamento dei risultati viene dunque conclusa e si procede con l'utilizzo della rete neurale sviluppata per la riproduzione dei BTD. Ad ogni modello vengono fornite rappresentazioni grafiche su piano cartesiano e tramite tabelle, per la visualizzazione dei risultati ottenuti.

5.3.1 Semplice appoggio 0/90/0

In questa sezione viene valutato il modello di struttura con vincolo del tipo semplice appoggio (S-S) su tutti i quattro lati. Lo spessore rimane costante e fissato ad a/h=10, mentre la curvatura viene fatta variare andando anche a considerare un caso piastra, ovvero in cui la curvatura risulta nulla.
In tabella 5.20 sono riportati i risultati ottenuti dall'analisi alle vibrazioni libere, dimostrando che quelli ottenuti dal codice di calcolo coincidono perfettamente con quelli già validati.

Tabella 5.20: 0/90/0, a/h=10, $\overline{\omega} = \omega\sqrt{\frac{\rho a^4}{h^2 E_T}}$.

Modello	R/a=2	R/a=5	Plate
TE 4	13.007	11.972	11.756
TE 4 [19]	13.007	11.972	11.756

Nelle tabelle 5.21, 5.22 e 5.23 sono invece riportati i best theory diagrams per ciascuno dei tre modelli considerati. Questi sono stati ottenuti tramite il codice di

calcolo strutturale considerando le combinazioni di tutti i dodici termini di espansione, come esposto in precedenza.

Tabella 5.21: BTD 0/90/0, a/h=10, R/a=2.

DOF	$u_{\alpha 1}$	$u_{\beta 1}$	u_{z1}	$u_{\alpha 2}$	$u_{\beta 2}$	u_{z2}	$u_{\alpha 3}$	$u_{\beta 3}$	u_{z3}	$u_{\alpha 4}$	$u_{\beta 4}$	u_{z4}	$u_{\alpha 5}$	$u_{\beta 5}$	u_{z5}
15	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲
14	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	△
13	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲	△	▲
12	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲	△	△
11	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	△	△	△
10	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	△	△	△	△
9	▲	▲	▲	▲	▲	▲	▲	△	▲	△	▲	△	△	△	△
8	▲	▲	▲	▲	▲	▲	△	△	▲	△	▲	△	△	△	△
7	▲	▲	▲	▲	▲	△	△	△	▲	△	▲	△	△	△	△
6	▲	▲	▲	▲	▲	△	△	△	△	△	▲	△	△	△	△
5	▲	▲	▲	▲	▲	△	△	△	△	△	△	△	△	△	△
4	▲	▲	▲	▲	△	△	△	△	△	△	△	△	△	△	△

Tabella 5.22: BTD 0/90/0, a/h=10, R/a=5.

DOF	$u_{\alpha 1}$	$u_{\beta 1}$	u_{z1}	$u_{\alpha 2}$	$u_{\beta 2}$	u_{z2}	$u_{\alpha 3}$	$u_{\beta 3}$	u_{z3}	$u_{\alpha 4}$	$u_{\beta 4}$	u_{z4}	$u_{\alpha 5}$	$u_{\beta 5}$	u_{z5}
15	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲
14	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲	▲	▲
13	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲	△	▲
12	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲	△	△
11	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	△	△	△
10	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	△	△	△	△
9	▲	▲	▲	▲	▲	▲	△	△	▲	▲	▲	△	△	△	△
8	▲	▲	▲	▲	▲	△	△	△	▲	▲	▲	△	△	△	△
7	▲	▲	▲	▲	▲	△	△	△	▲	△	▲	△	△	△	△
6	▲	▲	▲	▲	▲	△	△	△	△	△	▲	△	△	△	△
5	▲	▲	▲	▲	▲	△	△	△	△	△	△	△	△	△	△
4	▲	▲	▲	▲	△	△	△	△	△	△	△	△	△	△	△

Tabella 5.23: BTD 0/90/0, a/h=10, plate.

DOF	$u_{\alpha 1}$	$u_{\beta 1}$	u_{z1}	$u_{\alpha 2}$	$u_{\beta 2}$	u_{z2}	$u_{\alpha 3}$	$u_{\beta 3}$	u_{z3}	$u_{\alpha 4}$	$u_{\beta 4}$	u_{z4}	$u_{\alpha 5}$	$u_{\beta 5}$	u_{z5}
15	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲
14	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲	▲	▲	▲	▲
13	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲	▲	△	▲
12	▲	▲	▲	▲	▲	▲	△	△	▲	▲	▲	▲	△	▲	▲
11	▲	▲	▲	▲	▲	△	△	▲	▲	▲	▲	▲	△	△	▲
10	▲	▲	▲	▲	▲	△	△	△	▲	▲	▲	▲	△	△	▲
9	▲	▲	▲	▲	▲	△	△	△	▲	▲	▲	△	△	△	▲
8	▲	▲	▲	▲	▲	△	△	△	▲	▲	▲	△	△	△	△
7	▲	▲	▲	▲	▲	△	△	△	▲	△	▲	△	△	△	△
6	▲	▲	▲	▲	▲	△	△	△	△	△	▲	△	△	△	△
5	▲	▲	▲	▲	▲	△	△	△	△	△	△	△	△	△	△
4	▲	▲	▲	▲	△	△	△	△	△	△	△	△	△	△	△

La distribuzione dei termini ottenuta per ciascun modello analizzato, coincide con le distribuzioni presentate in [19], concludendo così il processo di accertamento dei risultati.
I dati ottenuti dal codice di calcolo vengono a questo punto fornite alla rete neurale artificiale sviluppata in questo capitolo, con le stesse modalità già presentate.
Nel caso in esame con curvatura R/a=2, è stata modificata la dimensione del layer di output, portandolo da 10 a 20, per osservare se la rete fosse in grado di fornire buoni risultati fino alla ventesima frequenza naturale del sistema.
A seguire vengono riportati i grafici cartesiani dei BTD di alcune frequenze naturali dei tre sistemi, mentre per una consultazione completa delle varie frequenze, si può trovare i rimanenti grafici in A.

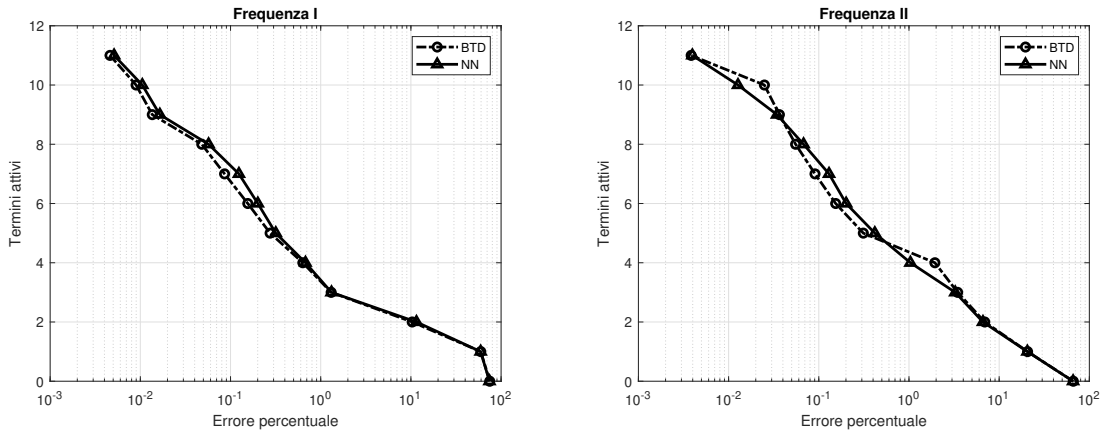


Figura 5.4: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze I e II.

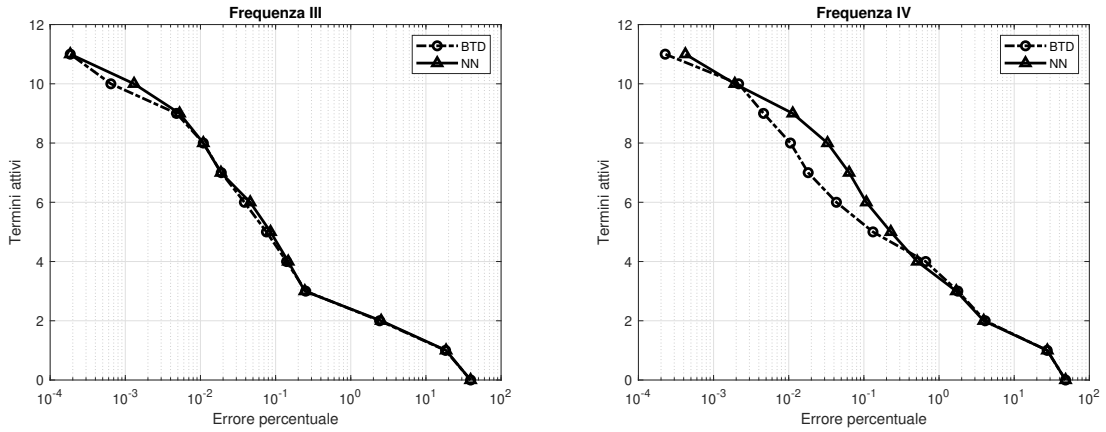


Figura 5.5: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze III e IV.

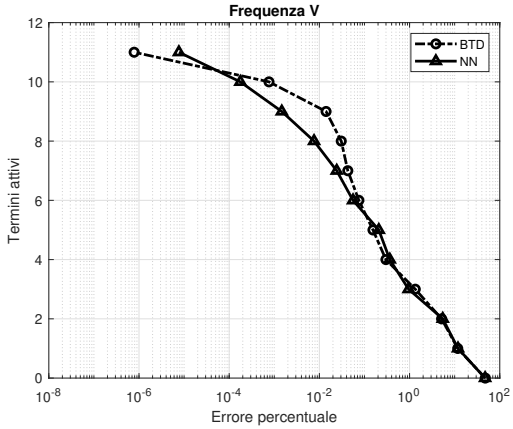


Figura 5.6: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenza V.

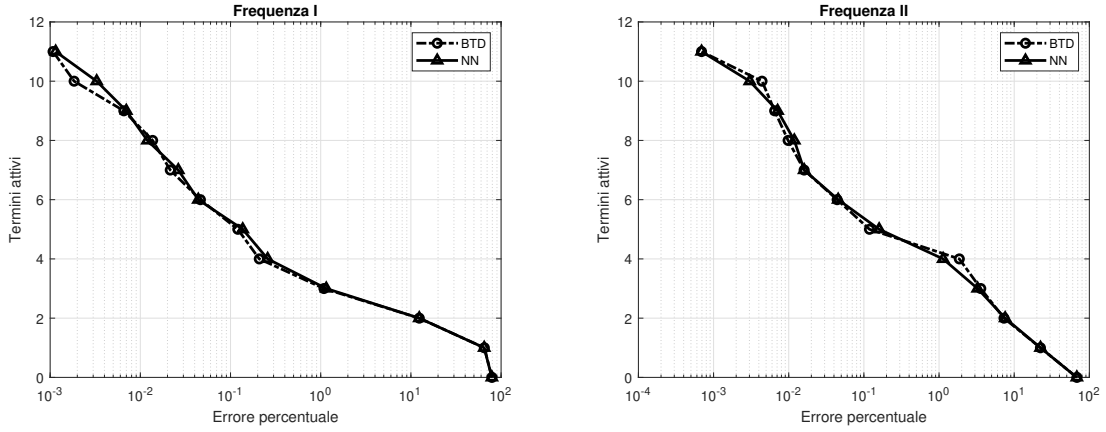


Figura 5.7: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5 - Frequenze I e II.

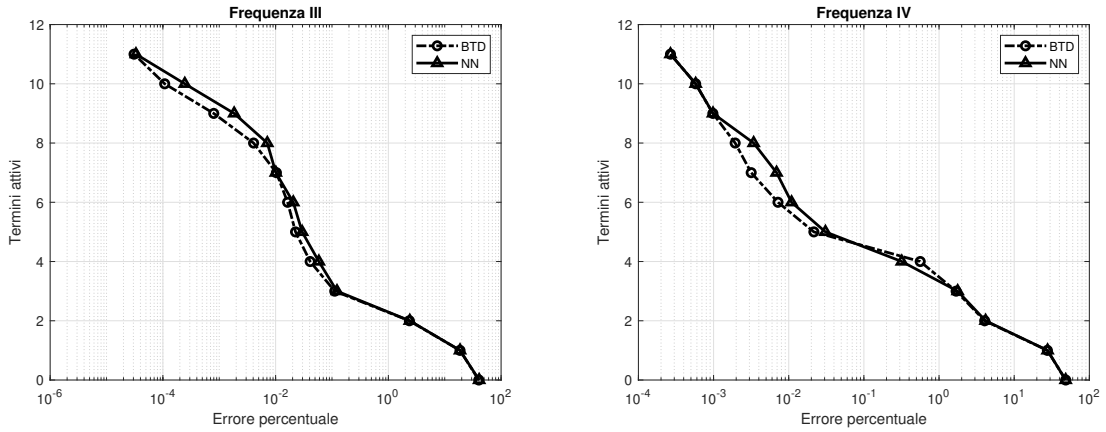


Figura 5.8: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5 - Frequenze III e IV.

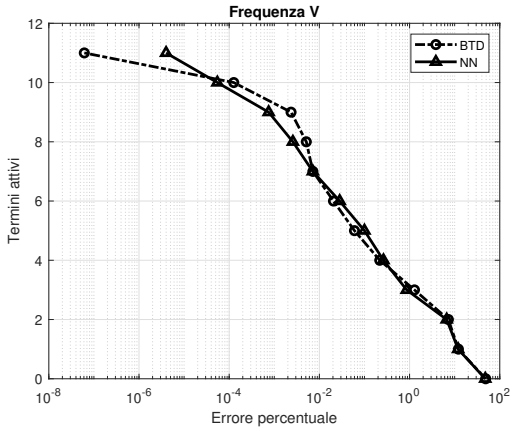


Figura 5.9: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5 - Frequenza V.

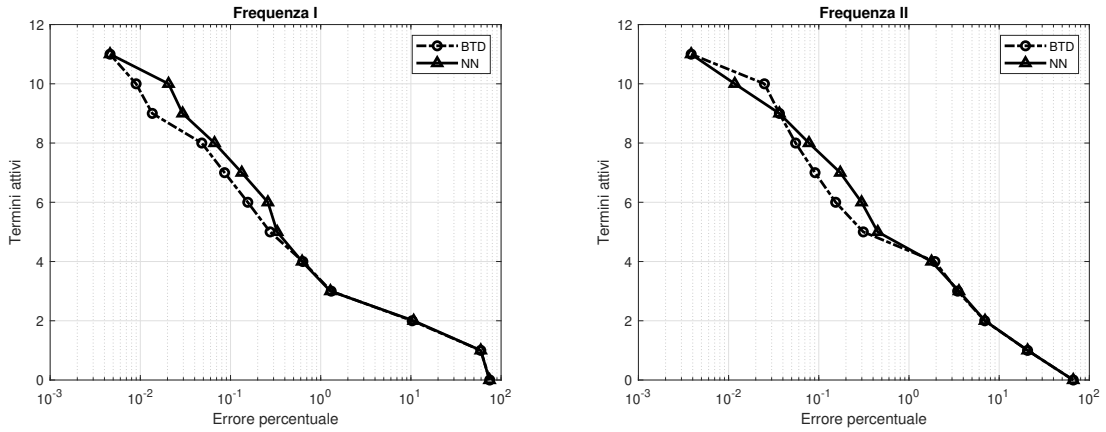


Figura 5.10: Piastra semplicemente appoggiata, 0/90/0, a/h=10 - Frequenze I e II.

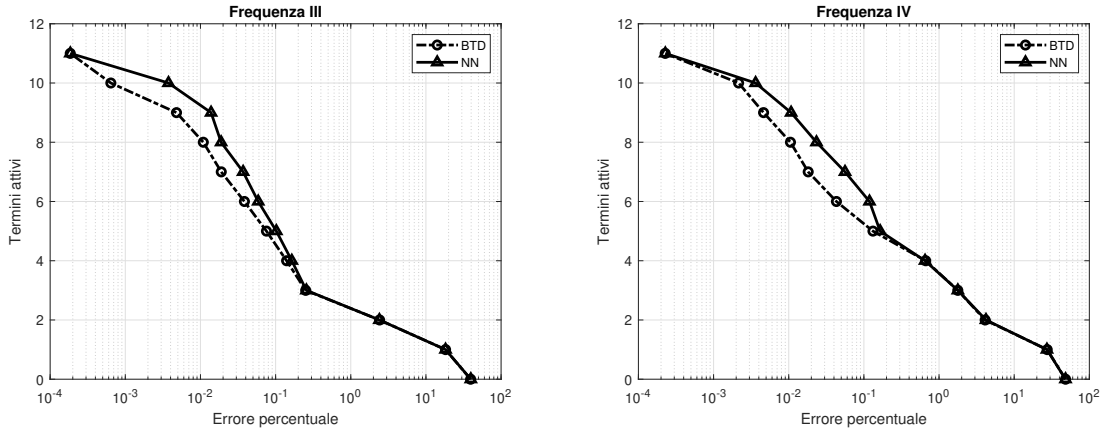


Figura 5.11: Piastra semplicemente appoggiata, 0/90/0, a/h=10 - Frequenze III e IV.

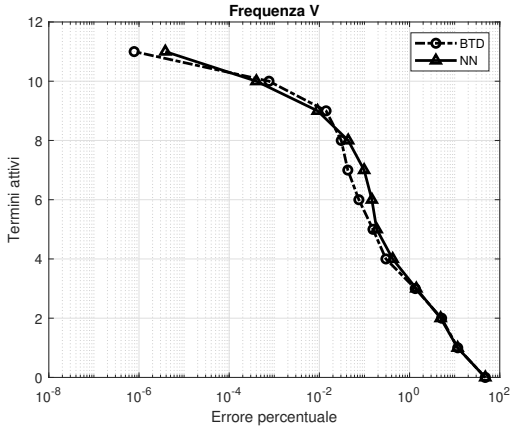


Figura 5.12: Piastra semplicemente appoggiata, 0/90/0, a/h=10 - Frequenza V.

Si osserva che l'errrore percentuale predetto dalla rete neurale in funzione del numero di termini attivi, segue molto fedelmente l'andamento di quanto calcolato dal software di calcolo. Alcuni punti si discostano più di altri, ma trovandoci in scala logaritmica si tratta in realtà di uno scostamento di un'ordine di grandezza esiguo e conseguentemente trascurabile, a favore della quantità di tempo nettamente inferiore con cui tale risultato è stato ottenuto rispetto alla sua controparte analitica.
Oltre a questi grafici in cui si può notare il confronto sull'errore in funzione dei termini attivi, vengono proposte le tabelle con le distribuzioni di termini attivi di ogni frequenza, per delle numerosità stabilite, a confronto con la media delle frequenze già riportata in precedenza nella fase di verifica dei risultati.

Tabella 5.24: BTD complessivo 5 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2.

Media					Frequenza I					Frequenza II					Frequenza III				
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	▲	△	△	△	▲	▲	△	△	△	▲	△	△	▲	△	▲	△	△	▲	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Frequenza IV					Frequenza V					Frequenza VI					Frequenza VII				
▲	▲	△	△	△	▲	▲	△	▲	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	△	△	▲	△	▲	△	△	△	△	▲	△	△	▲	△	▲	△	△	▲	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Frequenza VIII					Frequenza IX					Frequenza X					Frequenza XI				
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	▲	△	▲	▲	△	△	△
▲	△	△	▲	△	▲	△	△	▲	△	▲	△	△	△	△	▲	△	△	▲	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Frequenza XII					Frequenza XIII					Frequenza XIV					Frequenza XV				
▲	▲	△	△	△	▲	△	△	▲	△	▲	△	△	△	△	▲	△	△	△	△
▲	△	△	▲	△	▲	△	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Frequenza XVI					Frequenza XVII					Frequenza XVIII					Frequenza XIX				
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	△	△	▲	△	▲	△	△	▲	△	▲	△	▲	△	△	▲	△	▲	△	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Frequenza XX				
▲	▲	△	▲	△
▲	△	△	△	△
▲	△	△	△	△

Tabella 5.25: BTD complessivo 8 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2.

Media					Frequenza I					Frequenza II					Frequenza III				
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	▲	△	▲	▲	△	△	△
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	▲	▲	△	△	▲	▲	▲	△	△	▲	△	▲	△	△	▲	▲	▲	△	△
Frequenza IV					Frequenza V					Frequenza VI					Frequenza VII				
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	△	△
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	▲	▲	△	△
Frequenza VIII					Frequenza IX					Frequenza X					Frequenza XI				
▲	△	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	△	△	▲	△	▲	▲	△	▲	△	▲	△	△	▲	△	▲	▲	△	▲	△
▲	▲	△	△	△	▲	△	△	△	△	▲	△	▲	△	▲	▲	△	△	△	▲
Frequenza XII					Frequenza XIII					Frequenza XIV					Frequenza XV				
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	△	▲	△	△
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	△	▲	△	△	▲	△	▲	△	△	▲	△	▲	△	△	▲	▲	▲	△	△
Frequenza XVI					Frequenza XVII					Frequenza XVIII					Frequenza XIX				
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	▲	△	▲	▲	▲	▲	△	△
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	△	△	▲	▲	△	△	▲
▲	△	▲	△	△	▲	△	▲	△	△	▲	△	▲	△	△	▲	△	△	▲	△
Frequenza XX																			
▲	▲	△	▲	△															
▲	▲	△	▲	△															
▲	△	△	△	▲															

Tabella 5.26: BTD complessivo 12 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2.

Media					Frequenza I					Frequenza II					Frequenza III				
▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	▲
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	▲	▲	▲	△	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	△

Frequenza IV					Frequenza V					Frequenza VI					Frequenza VII				
▲	▲	▲	▲	△	▲	▲	△	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	▲
▲	▲	△	▲	▲	▲	▲	▲	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	▲	△	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	△

Frequenza VIII					Frequenza IX					Frequenza X					Frequenza XI				
▲	▲	▲	△	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	△	▲	▲	▲	▲	△
▲	△	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	△	▲	△
▲	▲	▲	△	▲	▲	▲	▲	△	△	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲

Frequenza XII					Frequenza XIII					Frequenza XIV					Frequenza XV				
▲	▲	△	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	△	▲
▲	▲	▲	▲	△	▲	▲	△	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲	▲
▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲	△	▲	▲	▲	△	△

Frequenza XVI					Frequenza XVII					Frequenza XVIII					Frequenza XIX				
▲	▲	▲	▲	△	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲
▲	▲	△	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	▲	▲	▲	▲	△	▲
▲	▲	▲	▲	▲	▲	▲	▲	△	△	▲	▲	▲	△	△	▲	▲	▲	▲	△

Frequenza XX				
▲	▲	▲	▲	△
▲	▲	▲	▲	△
▲	▲	▲	△	▲

Tabella 5.27: BTD complessivo 5 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5.

Media					Frequenza I					Frequenza II					Frequenza III				
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	▲	△	△	△	▲	▲	△	△	△	▲	△	△	▲	△	▲	△	△	▲	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Frequenza IV					Frequenza V					Frequenza VI					Frequenza VII				
▲	▲	△	△	△	▲	▲	△	▲	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	△	△	▲	△	▲	△	△	△	△	▲	△	△	▲	△	▲	△	△	▲	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Frequenza VIII					Frequenza IX					Frequenza X				
▲	▲	△	△	△	▲	▲	△	△	△	▲	△	△	▲	▲
▲	△	△	▲	△	▲	△	△	▲	△	▲	△	△	△	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Tabella 5.28: BTD complessivo 8 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5

Media					Frequenza I					Frequenza II					Frequenza III				
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	△	△
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	△	▲	△	△	▲	△	▲	△	△	▲	△	△	△	▲	▲	▲	△	△	▲

Frequenza IV					Frequenza V					Frequenza VI					Frequenza VII				
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	△	△
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	▲	△	△	▲

Frequenza VIII					Frequenza IX					Frequenza X				
▲	▲	▲	△	▲	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	△	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	▲	△	△	△	▲	△	△	△	▲	▲	△	△	△	▲

Tabella 5.29: BTD complessivo 12 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5.

Media					Frequenza I					Frequenza II					Frequenza III				
▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	▲	▲	△	▲	▲	△	▲	△
▲	▲	▲	▲	△	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲	▲

Frequenza IV					Frequenza V					Frequenza VI					Frequenza VII				
▲	▲	▲	▲	▲	▲	▲	△	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	▲
▲	▲	△	▲	△	▲	▲	▲	▲	▲	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	▲	△	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲

Frequenza VIII					Frequenza IX					Frequenza X				
▲	▲	▲	△	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	△
▲	▲	▲	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	▲	▲	▲	△	▲	▲	▲	△	▲	▲	▲	▲	▲	▲

Tabella 5.30: BTD complessivo 5 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10.

Media					Frequenza I					Frequenza II					Frequenza III				
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	▲	△	△	△	▲	▲	△	△	△	▲	△	△	▲	△	▲	△	△	▲	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Frequenza IV					Frequenza V					Frequenza VI					Frequenza VII				
▲	▲	△	△	△	▲	▲	△	▲	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	△	△	▲	△	▲	△	△	△	△	▲	△	△	▲	△	▲	△	△	▲	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Frequenza VIII					Frequenza IX					Frequenza X				
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	▲	△
▲	△	△	▲	△	▲	△	△	▲	△	▲	△	△	△	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Tabella 5.31: BTD complessivo 8 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10.

Media					Frequenza I					Frequenza II					Frequenza III				
▲	▲	△	▲	△	▲	▲	△	△	△	▲	▲	△	▲	△	▲	▲	△	△	△
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	△	▲	△	△	▲	▲	▲	△	△	▲	△	△	△	▲	▲	▲	△	△	▲
Frequenza IV					**Frequenza V**					**Frequenza VI**					**Frequenza VII**				
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	▲	△	△
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	▲	△	△	△
Frequenza VIII					**Frequenza IX**					**Frequenza X**									
▲	▲	△	▲	△	▲	▲	▲	△	▲	▲	▲	△	▲	△					
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	△	△	▲	△					
▲	△	△	△	▲	▲	△	△	△	△	▲	△	▲	△	▲					

Tabella 5.32: BTD complessivo 12 DOF - Piastra semplicemente appoggiata, 0/90/0, a/h=10.

Media					Frequenza I					Frequenza II					Frequenza III				
▲	▲	△	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	▲
▲	▲	△	▲	▲	▲	▲	▲	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	▲
▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲	▲	▲	△	△	▲
Frequenza IV					**Frequenza V**					**Frequenza VI**					**Frequenza VII**				
▲	▲	▲	▲	△	▲	▲	△	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	▲
▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	▲
▲	▲	△	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	△	△	▲
Frequenza VIII					**Frequenza IX**					**Frequenza X**									
▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	△					
▲	▲	△	▲	△	▲	▲	△	▲	▲	▲	▲	▲	▲	△					
▲	▲	▲	△	▲	▲	▲	▲	△	△	▲	▲	▲	▲	▲					

5.3.2 Clamped - Free 0/90/0

In questa sezione viene valutata l'influenza dello spessore, fissando il rapporto della curvatura ad R/a=10. Inoltre, in concomitanza alla variazione di spessore, viene utilizzato il tipo di vincolo clamped-free (C-F), coincidente con due lati del guscio liberi e gli altri due incastrati, in modo da aver lo stesso tipo di vincolo su lati opposti. Questa sezione viene presentata in maniera identica rispetto alla precedente. In tabella 5.33 sono riportati i risultati ottenuti dall'analisi dinamica, che anche in questo caso

coincidono con quelli già verificati.
Nelle tabelle 5.34 e 5.35 sono invece riportati i best theory diagrams ottenuti dal software di calcolo per i due modelli presi in esame e sono pure questi coincidenti con quelli già validati, potendo terminare a questo punto il processo di accertamento dei risultati.

Tabella 5.33: 0/90/0, R/a=10, $\overline{\omega} = \omega\sqrt{\frac{\rho a^4}{h^2 E_T}}$.

Modello	a/h=4	a/h=20
TE 4	7.250	24.144
TE 4 [19]	7.250	24.144

Tabella 5.34: BTD 0/90/0, R/a=10, a/h=4.

DOF	$u_{\alpha 1}$	$u_{\beta 1}$	u_{z1}	$u_{\alpha 2}$	$u_{\beta 2}$	u_{z2}	$u_{\alpha 3}$	$u_{\beta 3}$	u_{z3}	$u_{\alpha 4}$	$u_{\beta 4}$	u_{z4}	$u_{\alpha 5}$	$u_{\beta 5}$	u_{z5}
15	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲
14	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲
13	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	△	△	▲	▲
12	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	△	△	▲	▲
11	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	△	△	△	▲
10	▲	▲	▲	▲	▲	▲	△	△	▲	▲	▲	△	△	△	▲
9	▲	▲	▲	△	▲	▲	△	△	▲	▲	▲	△	△	△	▲
8	▲	▲	▲	△	▲	▲	△	△	▲	△	▲	△	△	△	▲
7	▲	▲	▲	△	▲	△	△	△	▲	△	▲	△	△	△	▲
6	▲	▲	▲	△	▲	▲	△	△	△	△	▲	△	△	△	△
5	▲	▲	▲	△	▲	△	△	△	△	△	▲	△	△	△	△
4	▲	▲	▲	△	△	△	△	△	△	△	▲	△	△	△	△

Tabella 5.35: BTD 0/90/0, R/a=10, a/h=20.

DOF	$u_{\alpha 1}$	$u_{\beta 1}$	u_{z1}	$u_{\alpha 2}$	$u_{\beta 2}$	u_{z2}	$u_{\alpha 3}$	$u_{\beta 3}$	u_{z3}	$u_{\alpha 4}$	$u_{\beta 4}$	u_{z4}	$u_{\alpha 5}$	$u_{\beta 5}$	u_{z5}
15	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲
14	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲
13	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	△	△	▲
12	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	△	△	▲
11	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	△	△	△	▲
10	▲	▲	▲	▲	▲	▲	△	△	▲	▲	▲	△	△	△	▲
9	▲	▲	▲	▲	▲	▲	△	△	▲	△	▲	△	△	△	▲
8	▲	▲	▲	▲	▲	▲	△	△	▲	△	▲	△	△	△	△
7	▲	▲	▲	△	▲	▲	△	△	▲	△	▲	△	△	△	△
6	▲	▲	▲	△	▲	△	△	△	▲	△	▲	△	△	△	△
5	▲	▲	▲	△	▲	△	△	△	△	△	▲	△	△	△	△
4	▲	▲	▲	△	▲	△	△	△	△	△	△	△	△	△	△

Analogamente a quanto è stato fatto per il caso di semplice appoggio, i dati ottenuti dal software di calcolo vengono forniti alla rete neurale. A seguire sono presentati i grafici cartesiani dei BTD così ottenuti per ognuna delle prime dieci frequenze naturali. La totalità delle rappresentazioni si può trovare in A.

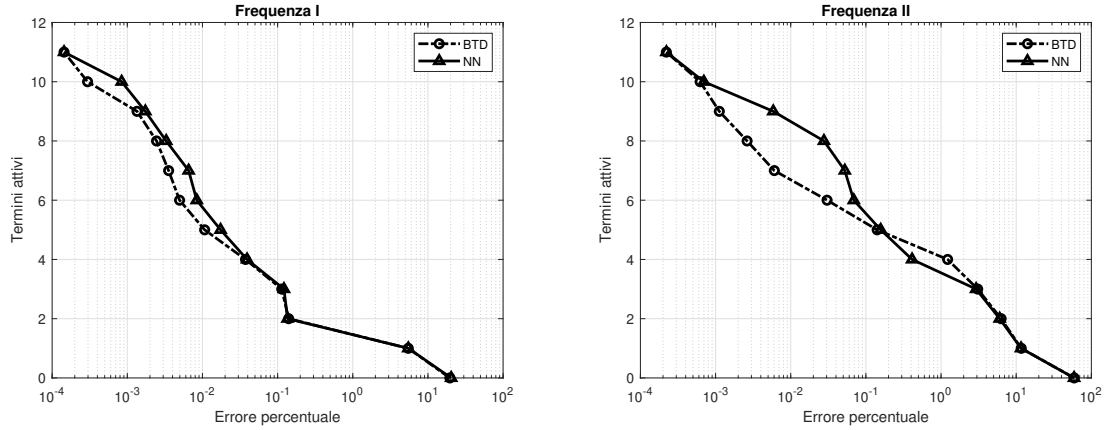


Figura 5.13: Piastra clamped - free, 0/90/0, a/h=4, R/a=10 - Frequenze I e II.

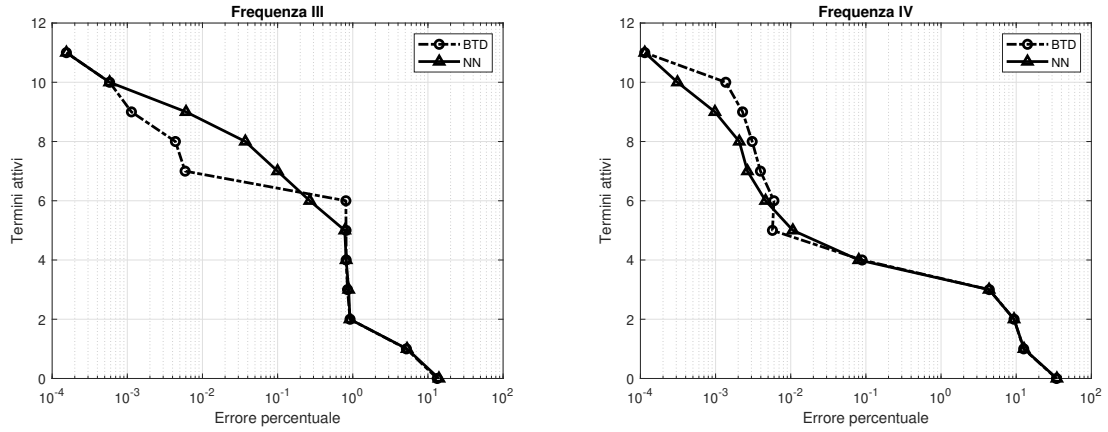


Figura 5.14: Piastra clamped - free, 0/90/0, a/h=4, R/a=10 - Frequenze III e IV.

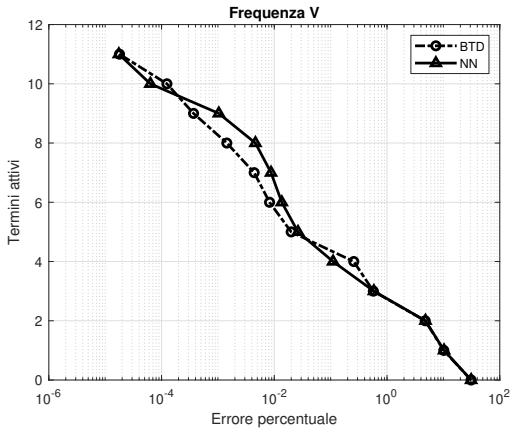


Figura 5.15: Piastra clamped - free, 0/90/0, a/h=4, R/a=10 - Frequenza V.

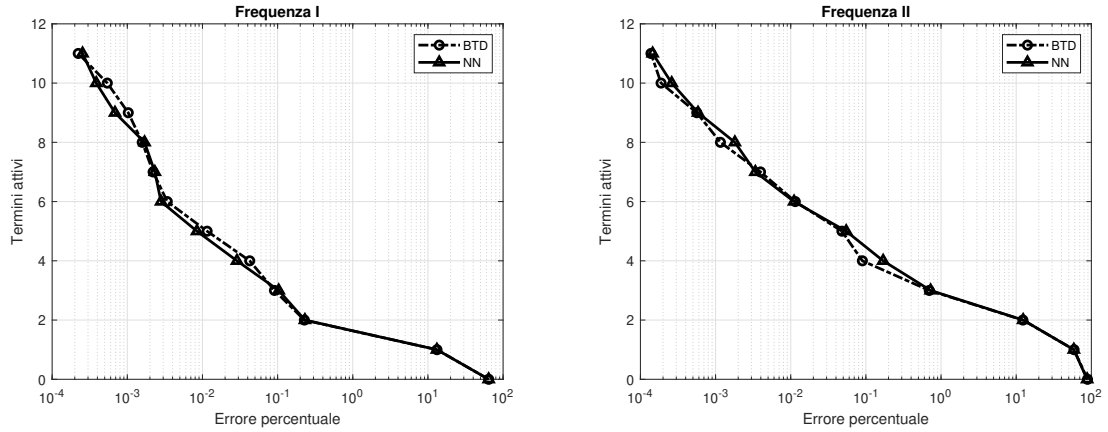


Figura 5.16: Piastra clamped - free, 0/90/0, a/h=20, R/a=10 - Frequenze I e II.

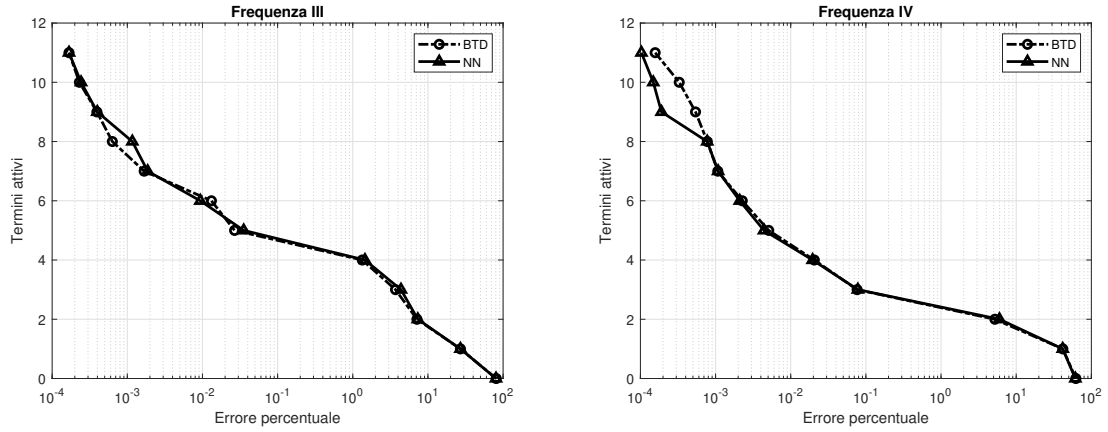


Figura 5.17: Piastra clamped - free, 0/90/0, a/h=20, R/a=10 - Frequenze III e IV.

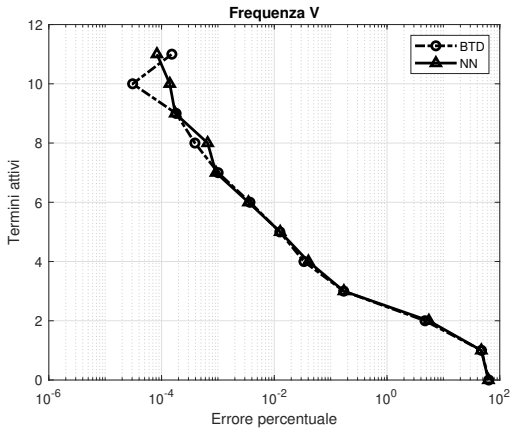


Figura 5.18: Piastra clamped - free, 0/90/0, a/h=20, R/a=10 - Frequenza V.

Anche nel caso clamped-free e con differenti spessori, la rete neurale riesce a predire l'andamento dell'errore percentuale in funzione del numero di termini attivi con un'ottima accuratezza. Si può notare che anche in questa casistica, i punti che risultano essere un po' più differenti tra btd ottenuto dal codice di calcolo e btd ottenuto dalla rete neurale sono comunque di ordine di grandezza irrisorio.
Vengono dunque proposte le tabelle in cui si possono osservare le distribuzioni di termini attivi, per alcune numerosità significative, e come queste cambino nelle varie frequenze.

Tabella 5.36: BTD complessivo 5 DOF - Piastra clamped/free, 0/90/0, R/a=10, a/h=4.

Media					Frequenza I					Frequenza II					Frequenza III				
▲	△	△	△	△	▲	△	△	△	△	▲	▲	△	△	△	▲	△	△	△	△
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	△	△	▲	△	▲	▲	△	▲	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Frequenza IV					Frequenza V					Frequenza VI					Frequenza VII				
▲	▲	△	△	△	▲	▲	△	△	△	▲	△	▲	△	▲	▲	▲	▲	△	△
▲	△	△	▲	△	▲	△	△	▲	△	▲	△	△	△	△	▲	△	△	△	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Frequenza VIII					Frequenza IX					Frequenza X				
▲	▲	▲	△	△	▲	△	△	▲	△	▲	▲	△	△	△
▲	△	△	△	△	▲	△	△	▲	△	▲	△	△	▲	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Tabella 5.37: BTD complessivo 8 DOF - Piastra clamped/free, 0/90/0, R/a=10, a/h=4.

Media					Frequenza I					Frequenza II					Frequenza III				
▲	△	△	△	△	▲	△	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	▲	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	△	▲	△	▲	△	△

Frequenza IV					Frequenza V					Frequenza VI					Frequenza VII				
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	△	▲	▲	▲	▲	▲	▲	▲	△
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	△	△	△	△	▲	▲	△	▲	△
▲	△	▲	△	△	▲	△	▲	△	△	▲	▲	△	▲	△	▲	△	△	△	△

Frequenza VIII					Frequenza IX					Frequenza X				
▲	▲	▲	△	△	▲	▲	▲	△	▲	▲	▲	▲	△	▲
▲	△	▲	△	△	▲	△	△	△	△	▲	▲	△	▲	△
▲	△	△	△	▲	▲	▲	△	▲	△	▲	△	△	△	△

Tabella 5.38: BTD complessivo 12 DOF - Piastra clamped/free, 0/90/0, R/a=10, a/h=4.

Media					Frequenza I					Frequenza II					Frequenza III				
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	▲	▲	△	▲	▲	△	▲	▲
▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	▲	▲	△	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲

Frequenza IV					Frequenza V					Frequenza VI					Frequenza VII				
▲	▲	▲	▲	△	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲
▲	▲	▲	▲	▲	▲	▲	△	▲	△	▲	△	▲	▲	▲	▲	▲	▲	▲	△
▲	△	▲	△	▲	▲	▲	▲	△	▲	▲	▲	△	▲	△	▲	△	▲	△	▲

Frequenza VIII					Frequenza IX					Frequenza X				
▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲
▲	▲	▲	▲	△	▲	△	▲	▲	▲	▲	▲	▲	▲	△
▲	△	▲	△	▲	▲	▲	△	▲	△	▲	△	▲	▲	▲

Tabella 5.39: BTD complessivo 5 DOF - Piastra clamped/free, 0/90/0, R/a=10, a/h=20.

Media					Frequenza I					Frequenza II					Frequenza III				
▲	△	△	△	△	▲	△	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	△	△	▲	△	△	▲	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Frequenza IV					Frequenza V					Frequenza VI					Frequenza VII				
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	△	△	▲	△	▲	△	△	▲	△	▲	△	△	▲	△	▲	△	△	▲	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Frequenza VIII					Frequenza IX					Frequenza X				
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	△	△	▲	△	▲	△	△	▲	△	▲	△	△	▲	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△

Tabella 5.40: BTD complessivo 8 DOF - Piastra clamped/free, 0/90/0, R/a=10, a/h=20.

Media					Frequenza I					Frequenza II					Frequenza III				
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	▲	▲	△	△	▲	▲	▲	△	△	▲	△	▲	△	△	▲	△	▲	△	△

Frequenza IV					Frequenza V					Frequenza VI					Frequenza VII				
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	△	▲	△	▲	△	△

Frequenza VIII					Frequenza IX					Frequenza X				
▲	▲	△	▲	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△
▲	△	▲	△	△	▲	△	▲	△	▲	▲	△	▲	△	▲

Tabella 5.41: BTD complessivo 12 DOF - Piastra clamped/free, 0/90/0, R/a=10, a/h=20.

Media					Frequenza I					Frequenza II					Frequenza III				
▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△
▲	▲	△	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△
▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲
Frequenza IV					**Frequenza V**					**Frequenza VI**					**Frequenza VII**				
▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△
▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△
▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲
Frequenza VIII					**Frequenza IX**					**Frequenza X**									
▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△					
▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△					
▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲					

Facendone un confronto si può osservare che nei due casi clamped-free, alcuni termini di ordine superiore vengono attivati più frequentemente rispetto ai casi di semplice appoggio, in cui compaiono termini di ordine inferiore. Un ulteriore commento deve essere fatto sull'influenza dello spessore, che risulta essere più marcata rispetto all'influenza data dalla curvatura. Infatti, aumentando il rapporto a/h si osserva che nelle teorie di ordine elevato, i termini che le compongono tendono a rimanere sempre gli stessi per tutte le dieci frequenze, cosa che non si verifica nelle altre casistiche.

Capitolo 6

Analisi Global/Local

In questo capitolo si vuole implementare la rete neurale artificale con output multidimensionale sviluppata in precedenza, per svolgere analisi strutturali del tipo global/local. Si vuole dunque ottenere una rete neurale in grado di fornire con accuratezza le migliori distribuzioni di teorie da utilizzare sugli elementi fem che compongono la mesh della struttura, a seconda del tipo di vincolo e dello spessore. Questo tipo di analisi risulta molto utile nel fine di ridurre i costi computazionali di calcolo, in quanto permette di stabilire l'utilizzo di modelli di teorie strutturali più accurati solamente in alcune zone della struttura, poiché risultano essere più critiche.
Il software di calcolo strutturale utilizzato finora permette di eseguire queste analisi global/local con metodo assiomatico-asintotico, distinguendo per ogni elemento fem tra due teorie strutturali. La soluzione proporrà dunque delle combinazioni tra la First-Order Shear Deformation Theory (FSDT) e quella ottenuta dalla completa espansione della funzione di Taylor di ordine quattro (N=4). Quest'ultima è inoltre, quando presente in ogni elemento fem, quella utilizzata come parametro di controllo per la valutazione dell'errore relativo prodotto dall'utilizzo delle varie combinazioni.
L'implementazione della rete neurale artificiale in questa casistica avviene nello stesso modo in cui era eseguita precedentemente. L'unica accortezza riguarda i dati relativi alle combinazioni in input. In questi casi vengono utilizzate mesh composte da 4x4 elementi, pertanto le combinazioni totali saranno pari a 2^{16}. Il parametro di controllo rimane l'errore percentuale sulle prime dieci frequenze naturali del sistema.
Come struttura in esame viene considerata un guscio di caratteristiche geometriche $a = b$, $R_\alpha = R_\beta = R$, rapporto $R/a = 5$. Le proprietà del materiale sono $E_1/E_2 = 25$, $G_{12}/E_2 = G_{13}/E_2 = 0.5$, $G_{13}/E_2 = 0.2$, $\nu = 0.25$. Le analisi vengono effettuate considerando un quarto di piastra, sfruttando le relazioni di simmetria, per poter ridurre ulteriormente il costo computazionale necessario.
Inizialmente si effettua una verifica del corretto funzionamento del codice di calcolo, riportata in tabella 6.1, andando a compiere un'analisi alle vibrazioni libere sulla struttura con un vincolo di semplice appoggio e si confronta il risultato ottenuto sulla prima frequenza con un valore già validato.

Dopo averne confermato la validità, si procede con la produzione dei risultati. Ad ogni casistica si effettua l'analisi global/local mediante software di calcolo strutturale e si utilizza l'output prodotto come input per la rete neurale artificiale, seguendo la metodologia già esposta. Il processo di analisi viene eseguito prima sulla media

Tabella 6.1: S-S, 0/90/0, a/h=10, R/a=5, $\overline{\omega} = \omega\sqrt{\frac{\rho a^4}{h^2 E_T}}$.

Modello	
TE 4	11.972
TE 4 [20]	11.972

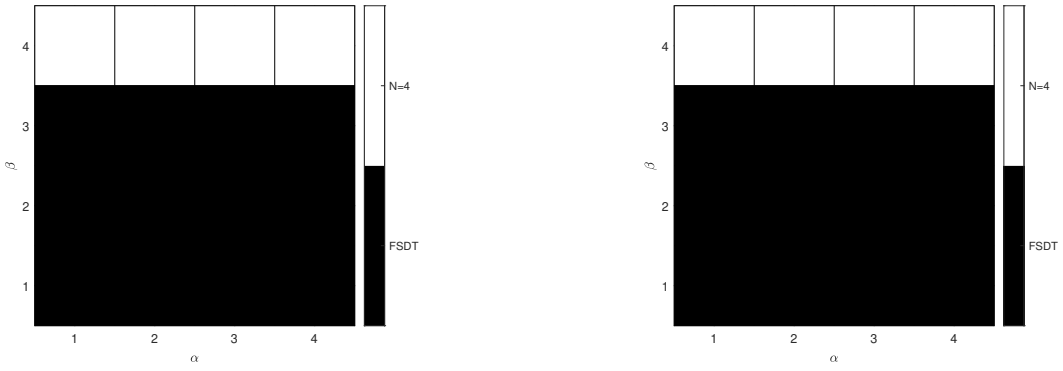
degli errori delle prime dieci frequenze naturali e successivamente su ognuna delle dieci frequenze. Nel caso della media viene riportata, in aggiunta ai risultati grafici, una tabella con con gli errori percentuali per ogni quantità di elementi N4 presenti, effettuando un confronto tra i valori ottenuti dal codice fem e quelli ottenuti dalla rete neurale. I risultati grafici sono invece riportati per le media e per ognuna delle prime cinque frequenze di ogni casistica, presentando un confronto tra distribuzione di elementi nel caso fem e nel caso rete neurale (NN), nelle configurazioni a quattro, otto e dodici elementi N4. Le rappresentazioni delle distribuzioni di teorie riguardanti le successive cinque frequenze naturali possono essere consultate in B.

6.1 S-S 0/90/0 a/h=10

La prima applicazione prevede l'utilizzo di un vincolo del tipo semplice appoggio (S-S), con uno spessore dal rapporto $a/h = 10$.

6.1.1 Media

Si osserva dalla tabella 6.2 che la rete neurale produce ottimi risultati, riuscendo a fornire una prediction dell'errore percentuale molto accurato e simile a quello calcolato tramite il software strutturale. Inoltre, esaminando le rappresentazioni di come le due differenti teorie sono distribuite sugli elementi della mesh, si riscontra un'analogia totale tra metodo fem e rete neurale artificiale.

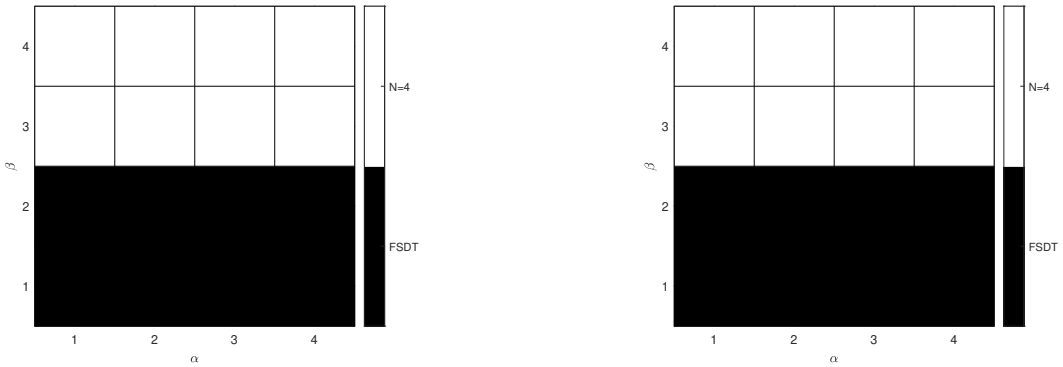


(a) FEM, Err=9.674%. (b) NN, Err=9.660%.

Figura 6.1: Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.

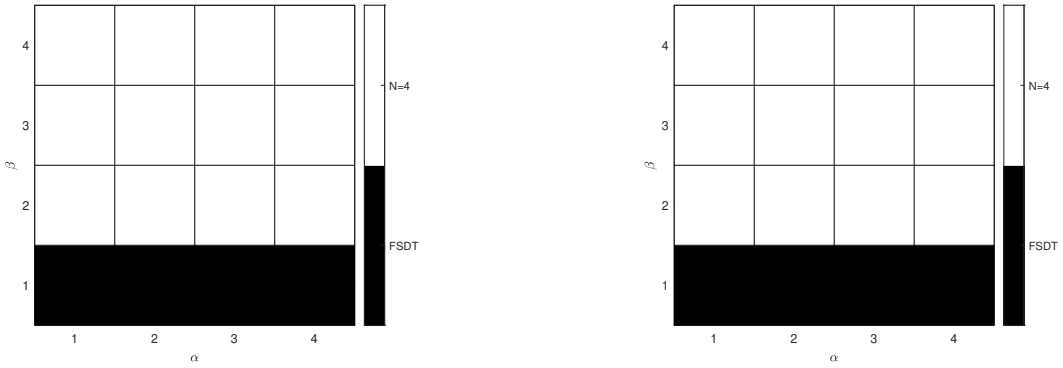
Tabella 6.2: S-S, 0/90/0, a/h=10, R/a=5.

El. N4	Err% Fem	Err% NN
1	12.360	12.360
2	11.678	11.671
3	10.839	10.862
4	9.674	9.660
5	9.030	9.056
6	8.372	8.380
7	7.721	7.720
8	6.769	6.798
9	6.100	6.074
10	5.340	5.281
11	4.523	4.478
12	3.595	3.698
13	2.436	3.160
14	2.174	2.591
15	1.437	1.437



(a) FEM, Err=6.769%. (b) NN, Err=6.798%.

Figura 6.2: Otto N4, S-S, 0/90/0, a/h=10, R/a=5.

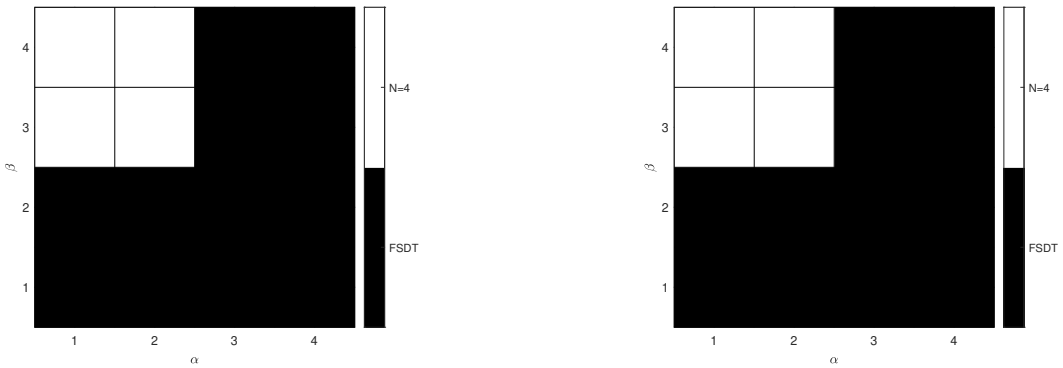


(a) FEM, Err=3.595%. (b) NN, Err=3.698%.

Figura 6.3: Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.

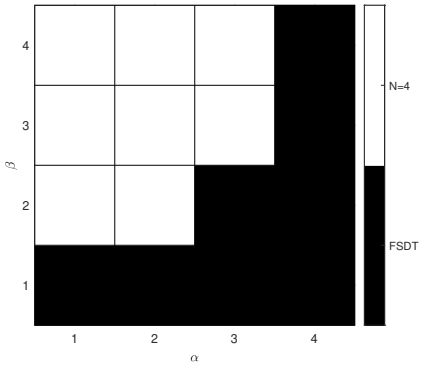
6.1.2 10 Frequenze

Vengono ora riportati i risultati ottenuti su ciascuna delle prime dieci frequenze naturali del sistema (le prime cinque sono riportate a seguire, le successive sono consultabili in B). Anche nel caso delle singole frequenze, si può notare l'accuratezza della rete neurale sulla valutazione dell'errore percentuale e sulla distribuzione degli elementi. In un numero ridotto di casi, la rete neurale non riproduce fedelmente la distribuzione ottenuta dal software di calcolo, identificando alcune teorie in elementi differenti, Questa variabilità si verifica in casistiche in cui la migliore distribuzione identificata dal metodo fem si discosta lievemente dal tipo di distribuzione identificata più uniformemente nelle altre frequenze. Inoltre, la non accuratezza di queste soluzioni fornite dalla rete neurale, possono essere anche causate dalla randomicità con cui vengono selezionati i modelli per la fase di training della rete, con la possibilità che quelli migliori non vi fossero inclusi.

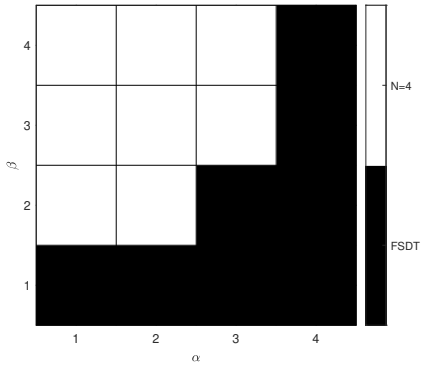


(a) FEM, Err=6.440%. (b) NN, Err=6.440%.

Figura 6.4: Frequenza I - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.

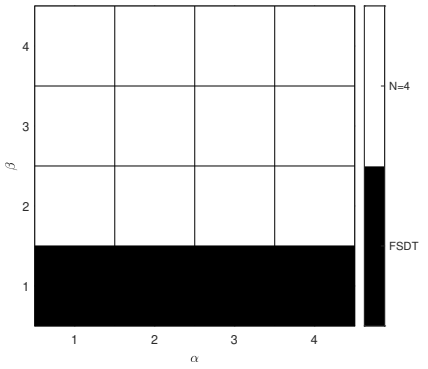


(a) FEM, Err=2.538%.

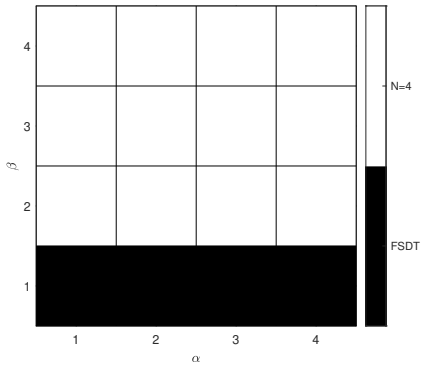


(b) NN, Err=2.513%.

Figura 6.5: Frequenza I - Otto N4, S-S, 0/90/0, a/h=10, R/a=5.

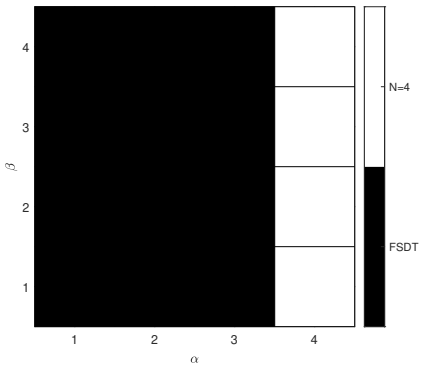


(a) FEM, Err=0.570%.

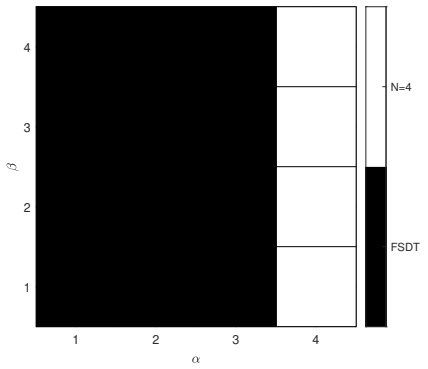


(b) NN, Err=0.530%.

Figura 6.6: Frequenza I - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.

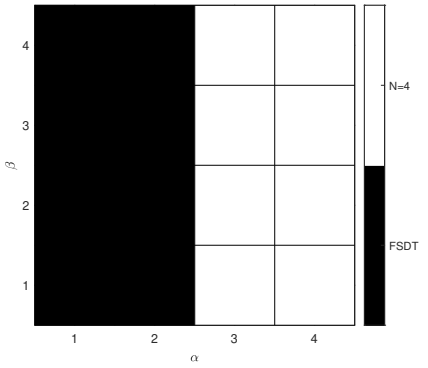


(a) FEM, Err=5.939%.

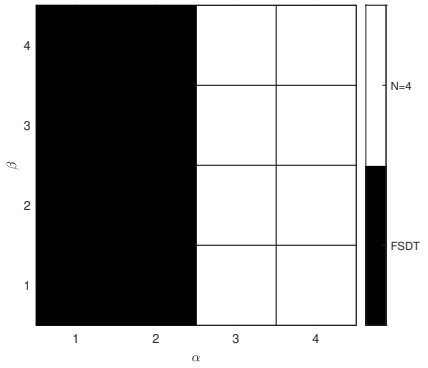


(b) NN, Err=5.939%.

Figura 6.7: Frequenza II - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.

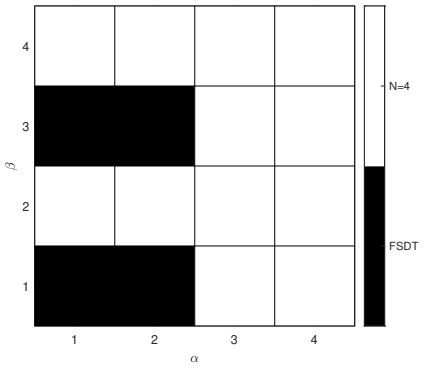


(a) FEM, Err=3.614%.

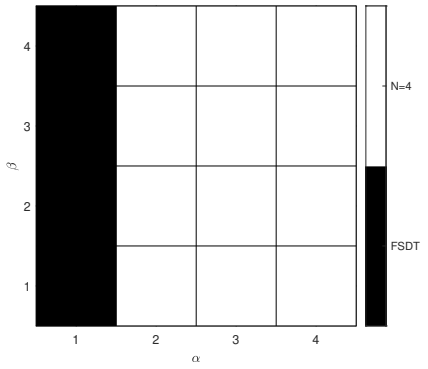


(b) NN, Err=3.581%.

Figura 6.8: Frequenza II - Otto N4, S-S, 0/90/0, a/h=10, R/a=5.

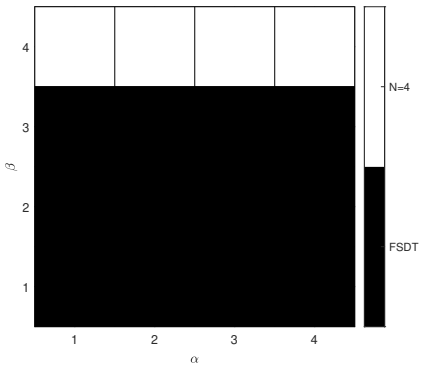


(a) FEM, Err=1.726%.

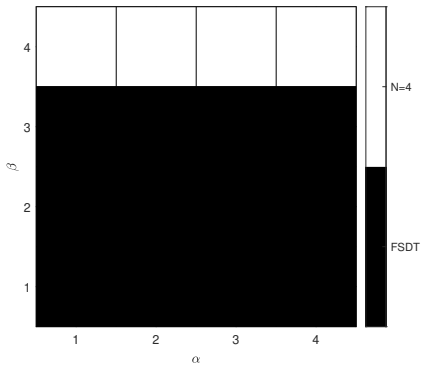


(b) NN, Err=1.760%.

Figura 6.9: Frequenza II - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.

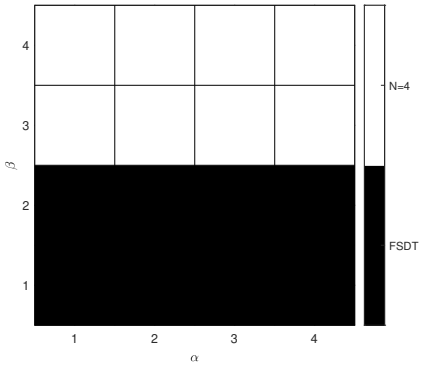


(a) FEM, Err=12.026%.

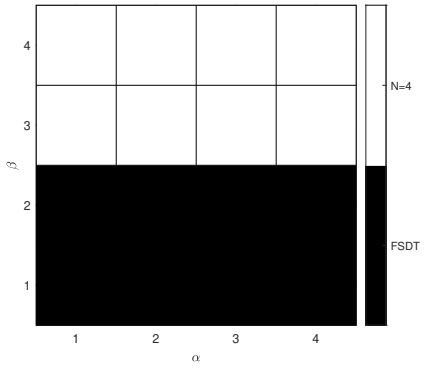


(b) NN, Err=12.095%.

Figura 6.10: Frequenza III - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.

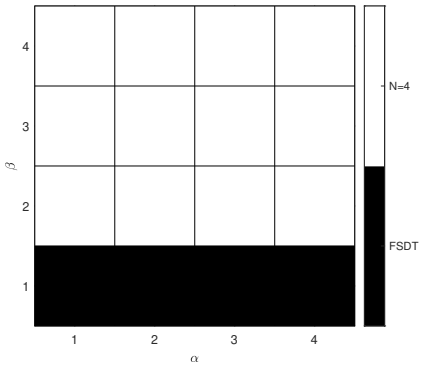


(a) FEM, Err=4.036%.

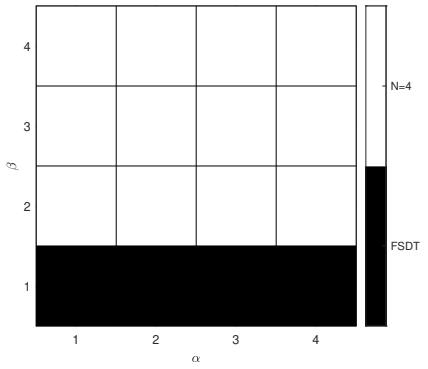


(b) NN, Err=4.053%.

Figura 6.11: Frequenza III - Otto N4, S-S, 0/90/0, a/h=10, R/a=5.

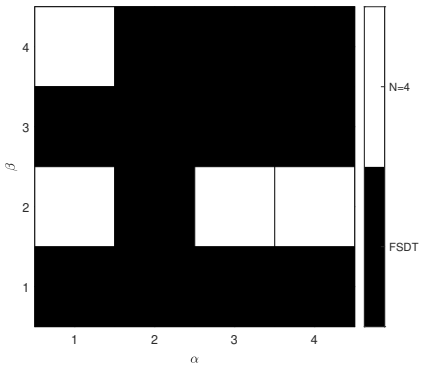


(a) FEM, Err=0.795%.

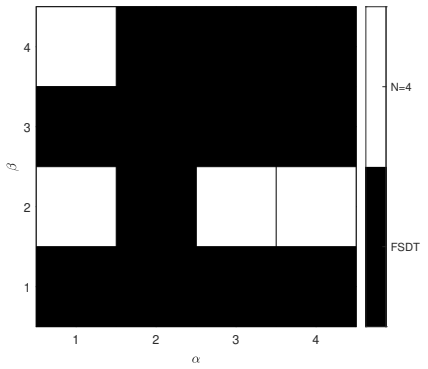


(b) NN, Err=0.744%.

Figura 6.12: Frequenza III - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.

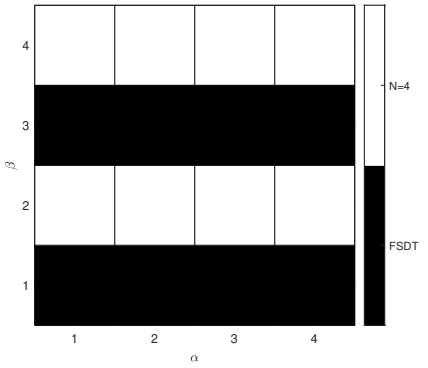


(a) FEM, Err=12.739%.

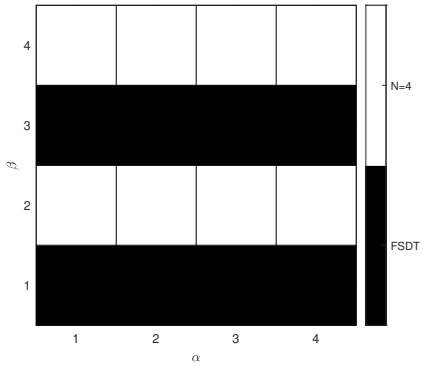


(b) NN, Err=12.717%.

Figura 6.13: Frequenza IV - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.

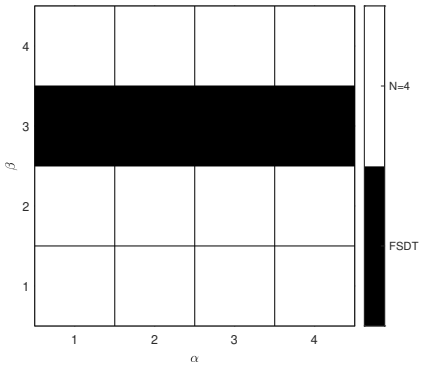


(a) FEM, Err=7.452%.

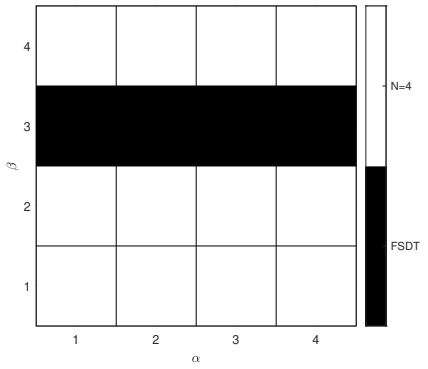


(b) NN, Err=7.590%.

Figura 6.14: Frequenza IV - Otto N4, S-S, 0/90/0, a/h=10, R/a=5.

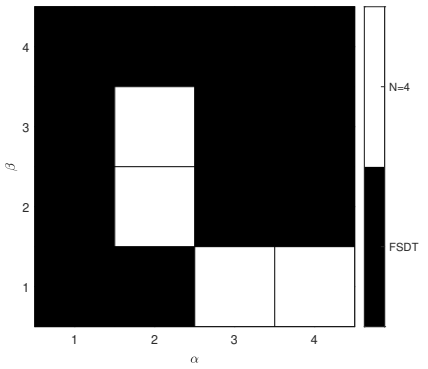


(a) FEM, Err=2.834%.

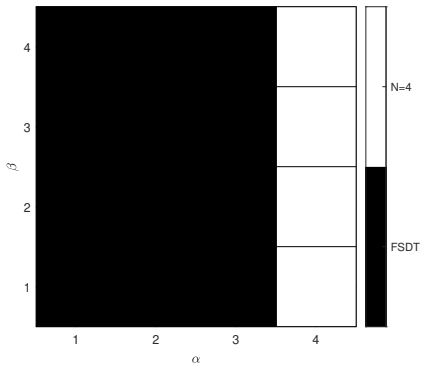


(b) NN, Err=2.827%.

Figura 6.15: Frequenza IV - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.

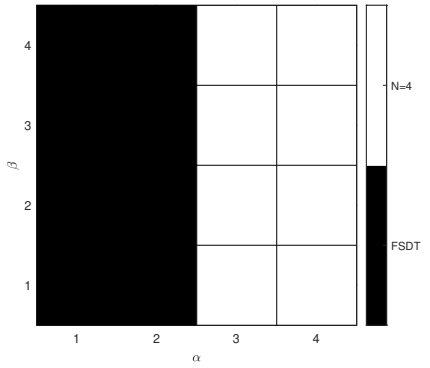


(a) FEM, Err=5.238%.

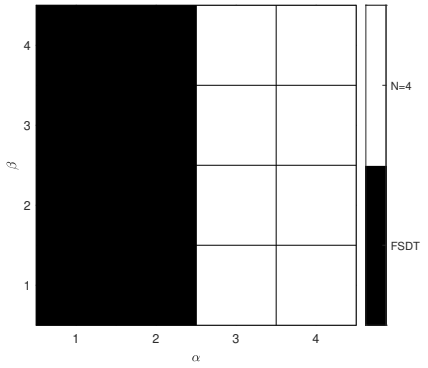


(b) NN, Err=5.758%.

Figura 6.16: Frequenza V - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.

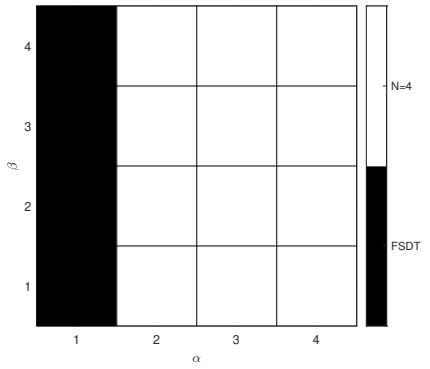


(a) FEM, Err=2.550%.

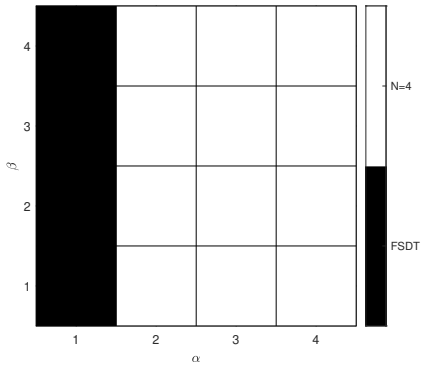


(b) NN, Err=2.612%.

Figura 6.17: Frequenza V - Otto N4, S-S, 0/90/0, a/h=10, R/a=5.



(a) FEM, Err=0.845%.



(b) NN, Err=0.766%.

Figura 6.18: Frequenza V - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.

6.2 S-S 0/90/0 a/h=10 - Dati training ridotti

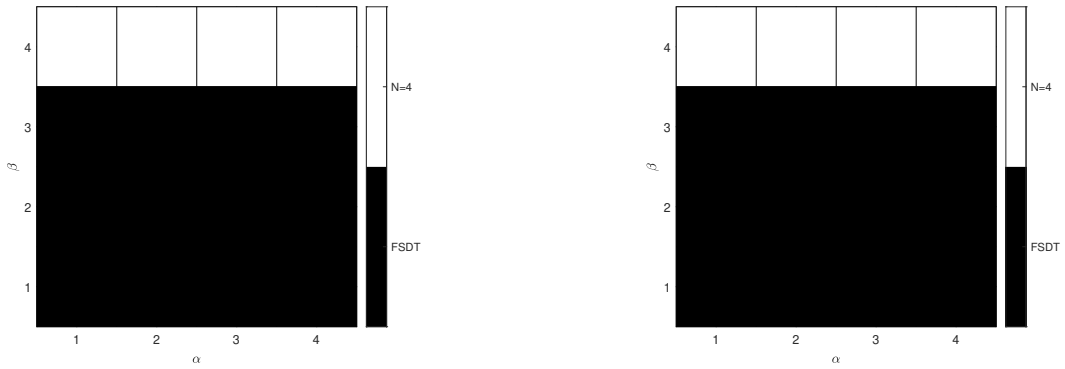
In questa sezione si è voluto sperimentare le potenzialità della rete neurale (NN) andando ad osservare l'impatto sui risultati che poteva avere la riduzione percentuale dei dati in input forniti alla fase di training. La struttura quindi è soggetta allo stesso tipo di vincolo della sezione precedente e presenta lo stesso spessore. Le combinazioni totali prodotte dal software di calcolo sono sempre 2^{16}, ma stavolta ne sono stati selezionati il 10%, anziché il 20%, per la fase di training della rete neurale artificiale. La metodologia di selezione non ha subito variazioni, quello che è stato eseguito è la divisione a metà dei dati selezionati da ogni numerosità di elementi. Riepilogando, in questa sezione vengono dunque utilizzati un 5% dei dati totali per il training della rete ed un altro 5% per la sua validazione.

6.2.1 Media

Si può osservare dai valori riportati in tabella 6.3, che la riduzione dei dati utilizzati all'interno della rete neurale non ha portato cambiamenti significativi nell'accuratezza, alcuni risultati sono migliorati rispetto ai precedenti, mentre alcuni sono peggiorati, ma comunque in un intervallo esiguo. Inoltre, anche le distribuzioni delle teorie sugli elementi risultano analoghe.

Tabella 6.3: S-S, 0/90/0, a/h=10, R/a=5, 10% dati input nn.

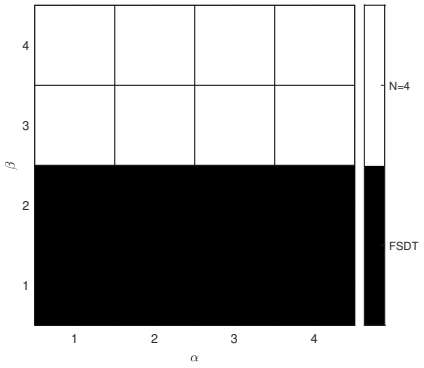
El. N4	Err% Fem	Err% NN
1	12.360	12.360
2	11.678	11.722
3	10.839	10.840
4	9.674	9.673
5	9.030	9.018
6	8.372	8.343
7	7.721	7.691
8	6.769	6.760
9	6.100	6.124
10	5.340	5.306
11	4.523	4.519
12	3.595	3.560
13	2.436	3.190
14	2.174	2.319
15	1.437	1.437



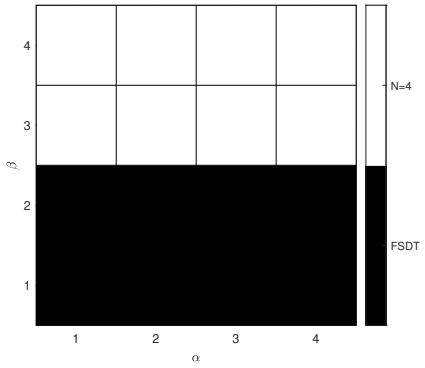
(a) FEM, Err=9.674%.

(b) NN, Err=9.673%.

Figura 6.19: Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.

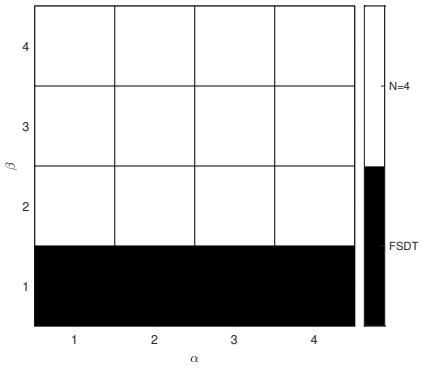


(a) FEM, Err=6.769%.

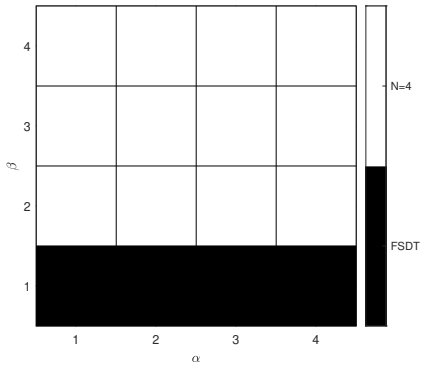


(b) NN, Err=6.760%.

Figura 6.20: Otto N4, S-S, 0/90/0, a/h=10, R/a=5.



(a) FEM, Err=3.595%.

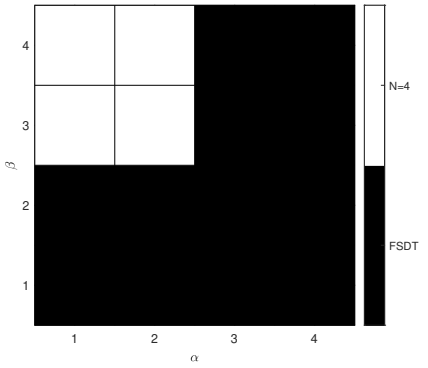


(b) NN, Err=3.560%.

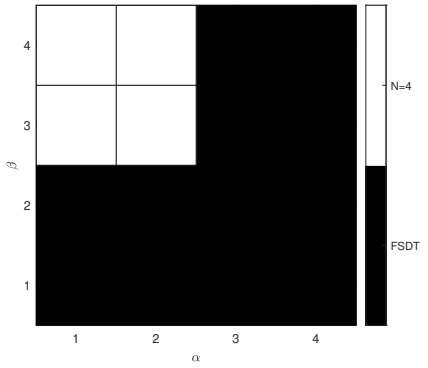
Figura 6.21: Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.

6.2.2 10 Frequenze

Anche in questo caso le distribuzioni di modelli di teorie individuate dalla rete neurale non sono coincidenti con quelle fem solo in alcuni casi di alcune frequenze. Questa diversità per di più, si riscontra negli stessi modelli che si verificava con l'utilizzo del 20% dei dati totali per il training della rete. Si può quindi effettuare un'importante osservazione sul fatto che la rete neurale sviluppata continua a rimanere accurata anche con un'esigua parte di dati ad essa fornita, confermandone le sue ottime potenzialità.

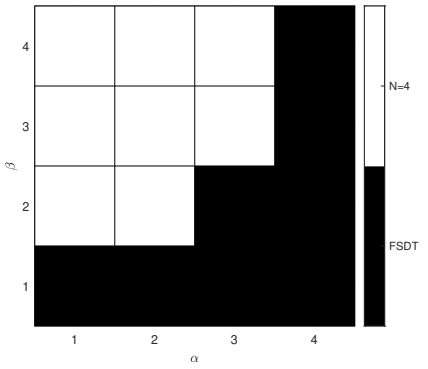


(a) FEM, Err=6.440%.

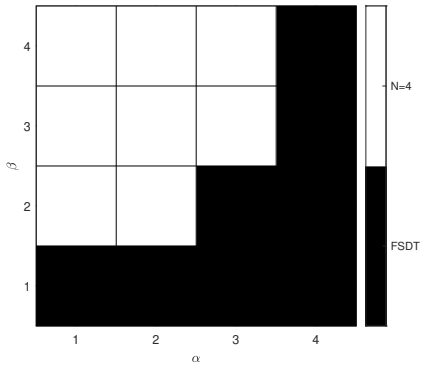


(b) NN, Err=6.456%.

Figura 6.22: Frequenza I - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.

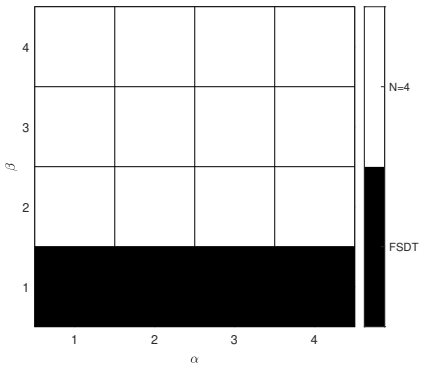


(a) FEM, Err=2.538%.

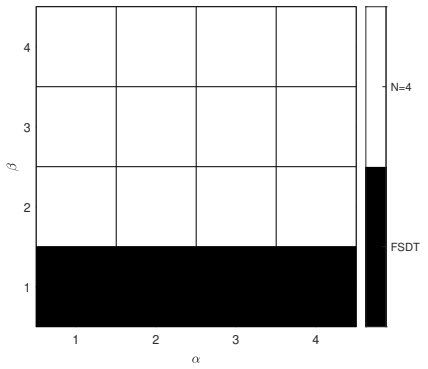


(b) NN, Err=2.505%.

Figura 6.23: Frequenza I - Otto N4, S-S, 0/90/0, a/h=10, R/a=5.

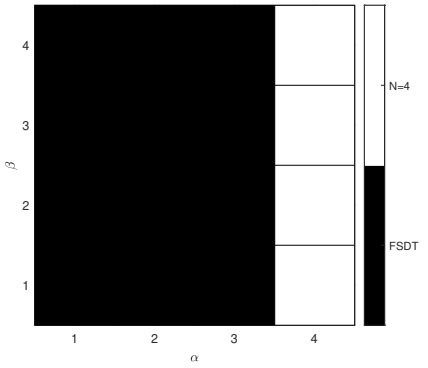


(a) FEM, Err=0.570%.

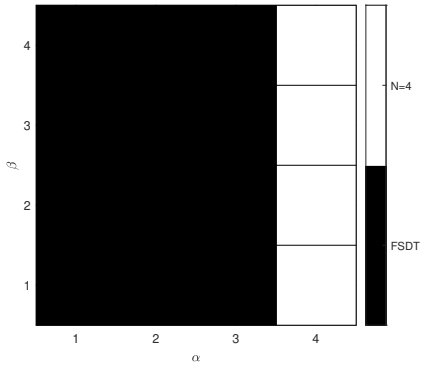


(b) NN, Err=0.548%.

Figura 6.24: Frequenza I - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.

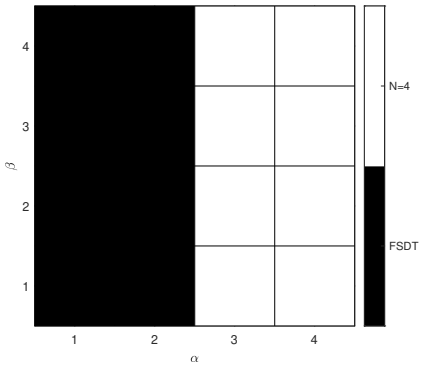


(a) FEM, Err=5.939%.

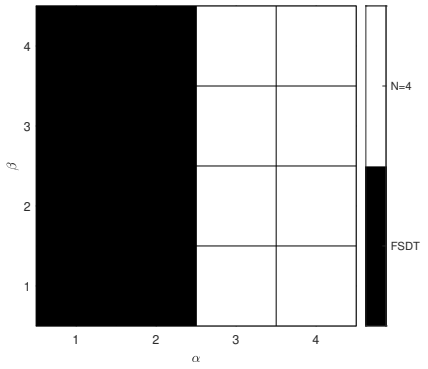


(b) NN, Err=5.946%.

Figura 6.25: Frequenza II - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.

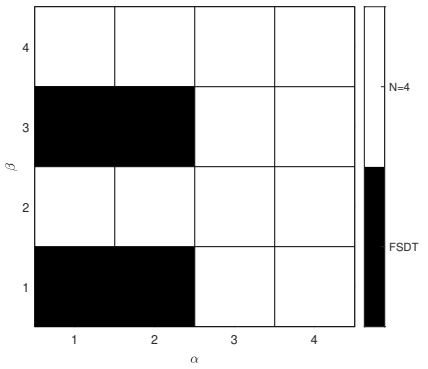


(a) FEM, Err=3.614%.

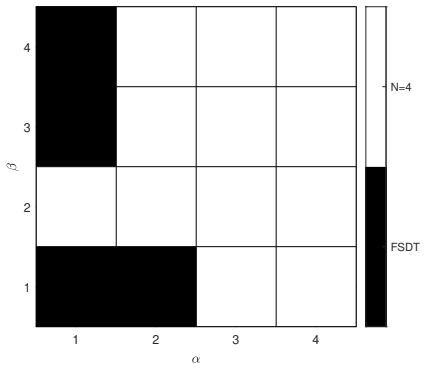


(b) NN, Err=3.630%.

Figura 6.26: Frequenza II - Otto N4, S-S, 0/90/0, a/h=10, R/a=5.

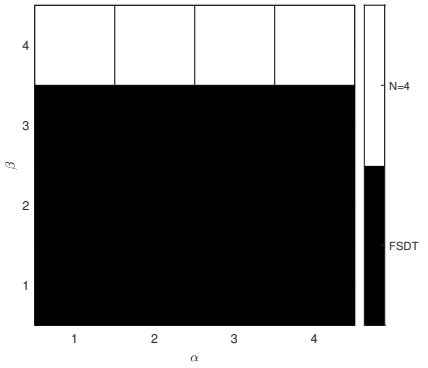


(a) FEM, Err=1.726%.

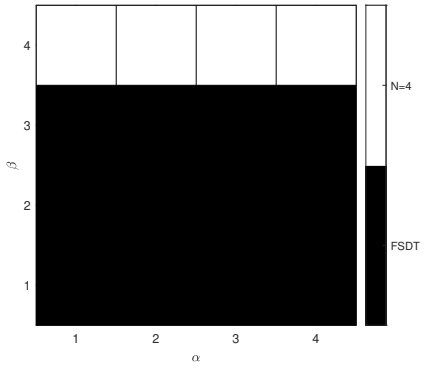


(b) NN, Err=1.721%.

Figura 6.27: Frequenza II - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.

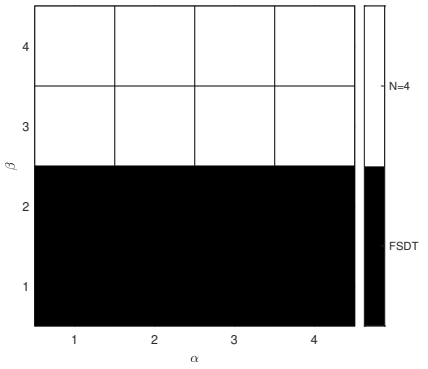


(a) FEM, Err=12.026%.

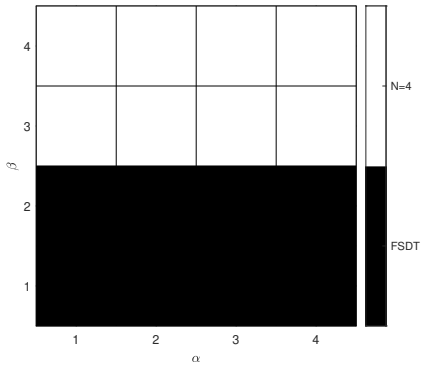


(b) NN, Err=12.030%.

Figura 6.28: Frequenza III - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.

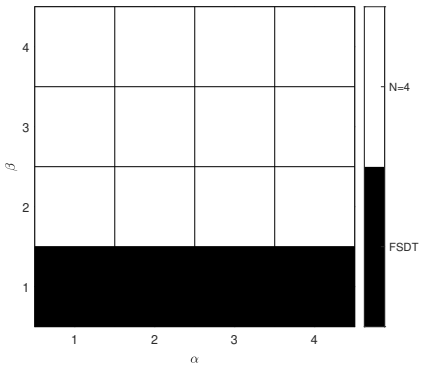


(a) FEM, Err=4.036%.

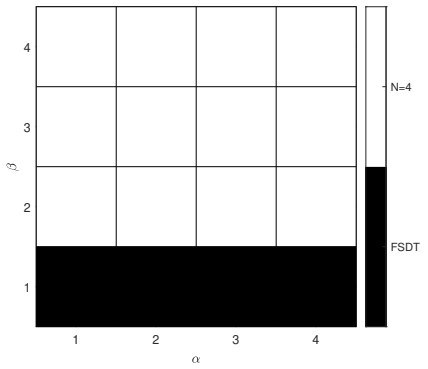


(b) NN, Err=4.125%.

Figura 6.29: Frequenza III - Otto N4, S-S, 0/90/0, a/h=10, R/a=5.

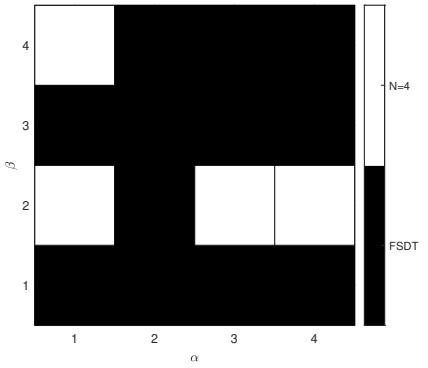


(a) FEM, Err=0.795%.

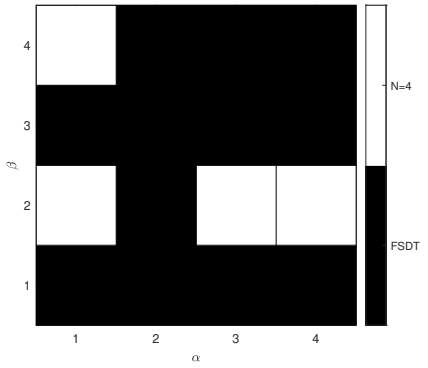


(b) NN, Err=0.877%.

Figura 6.30: Frequenza III - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.

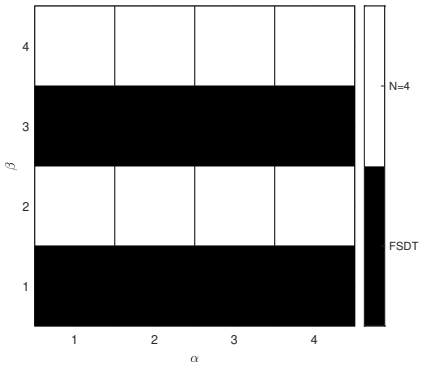


(a) FEM, Err=12.739%.

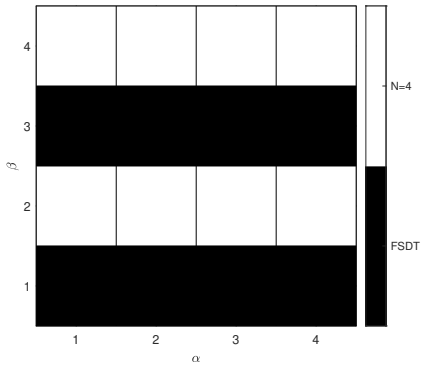


(b) NN, Err=12.718%.

Figura 6.31: Frequenza IV - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.

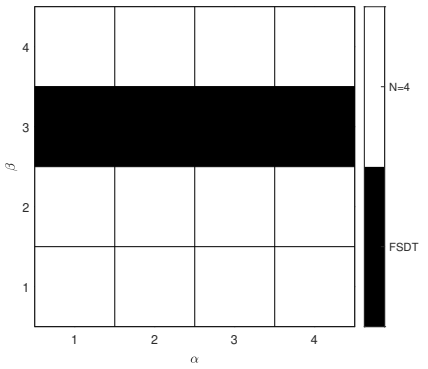


(a) FEM, Err=7.452%.

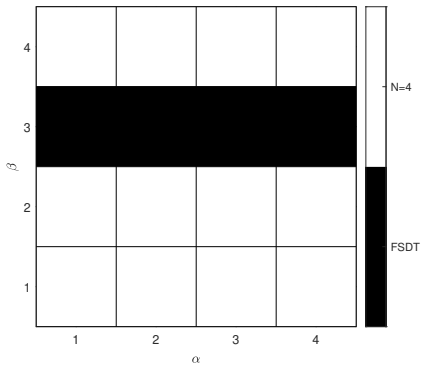


(b) NN, Err=7.439%.

Figura 6.32: Frequenza IV - Otto N4, S-S, 0/90/0, a/h=10, R/a=5.

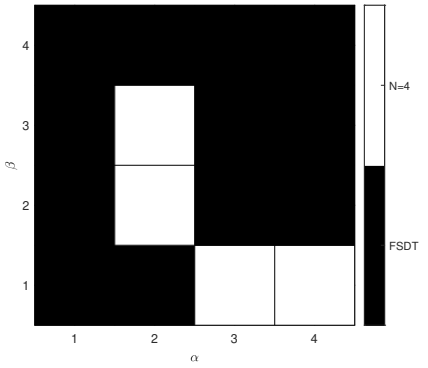


(a) FEM, Err=2.834%.

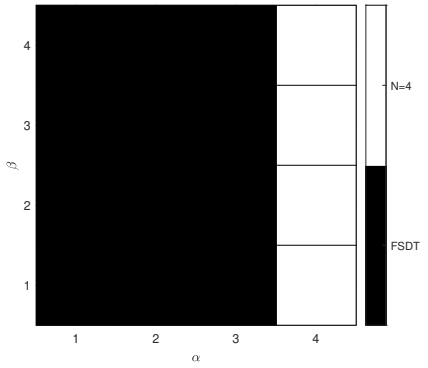


(b) NN, Err=3.222%.

Figura 6.33: Frequenza IV - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.

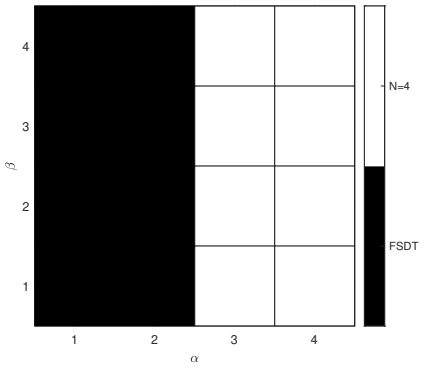


(a) FEM, Err=5.238%.

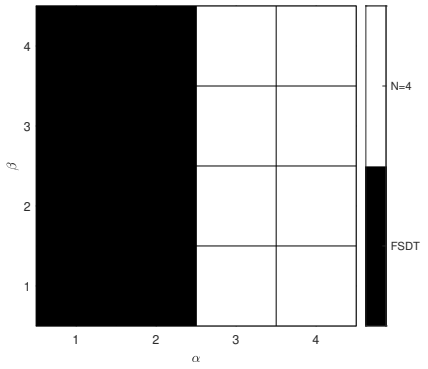


(b) NN, Err=5.802%.

Figura 6.34: Frequenza V - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.

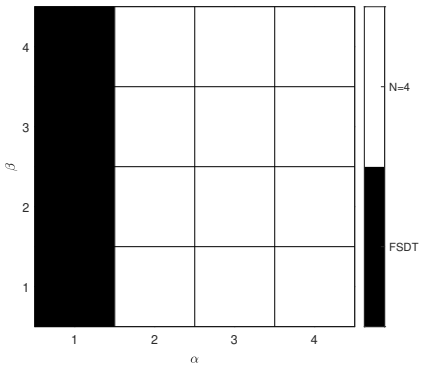


(a) FEM, Err=2.550%.

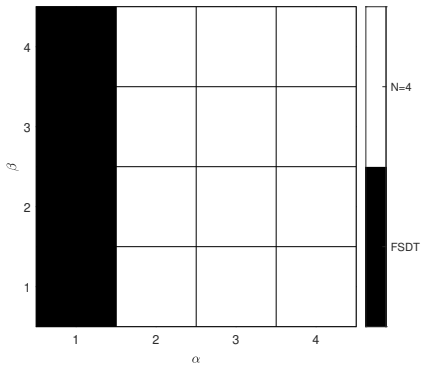


(b) NN, Err=2.534%.

Figura 6.35: Frequenza V - Otto N4, S-S, 0/90/0, a/h=10, R/a=5.



(a) FEM, Err=0.845%.



(b) NN, Err=0.821%.

Figura 6.36: Frequenza V - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.

6.3 S-S 0/90/0 a/h=100

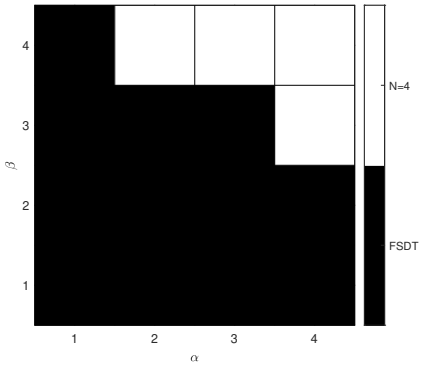
Viene ora proposta una variazione dello spessore della struttura, in particolare il rapporto di spessore utilizzato per questa casistica è $a/h = 100$. Le condizoni di vincolo rimangono quelle del semplice appoggio.

6.3.1 Media

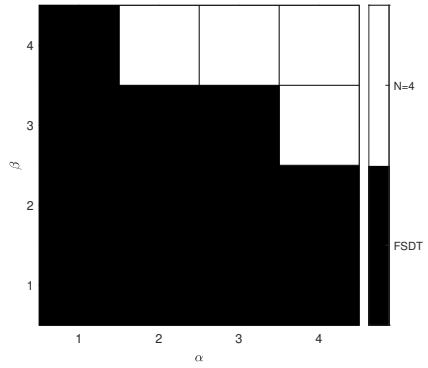
Nella tabella 6.4 si osserva il confronto tra risultati prodotti dal codice di calcolo strutturale e quelli predetti dalla rete neurale artificiale. Diminuendo lo spessore della struttura in esame, la rete neurale continua a fornire ottimi risultati, con un'accuratezza paragonabile a quella per uno spessore maggiore, risultando per alcuni casi anche più precisa. La riproduzione della distribuzione dei modelli di teorie rimane fedele a quella fem.

Tabella 6.4: S-S, 0/90/0, a/h=100, R/a=5.

El. N4	Err% Fem	Err% NN
1	3.011	3.011
2	2.779	2.779
3	2.521	2.516
4	2.307	2.307
5	2.059	2.061
6	1.840	1.841
7	1.644	1.640
8	1.453	1.435
9	1.217	1.217
10	1.000	1.002
11	0.806	0.796
12	0.628	0.618
13	0.442	0.441
14	0.298	0.295
15	0.127	0.128

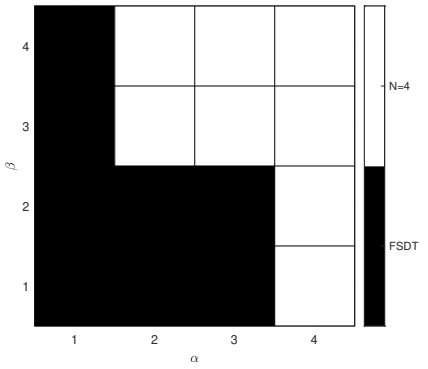


(a) FEM, Err=2.059%.

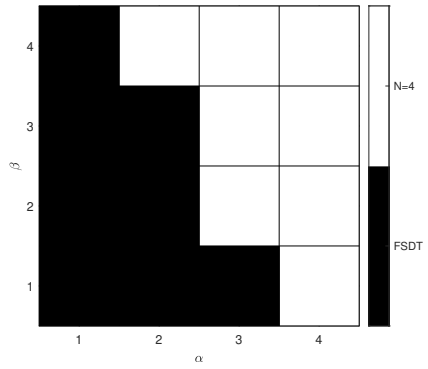


(b) NN, Err=2.061%.

Figura 6.37: Quattro N4, S-S, 0/90/0, a/h=100, R/a=5.

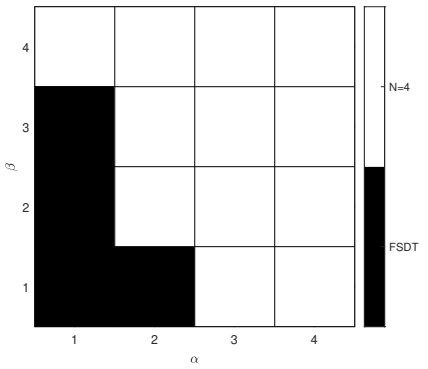


(a) FEM, Err=1.453%.

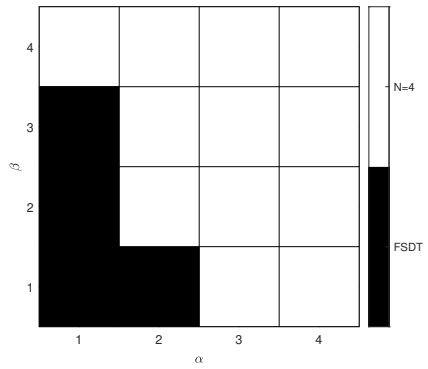


(b) NN, Err=1.435%.

Figura 6.38: Otto N4, S-S, 0/90/0, a/h=100, R/a=5.



(a) FEM, Err=0.628%.

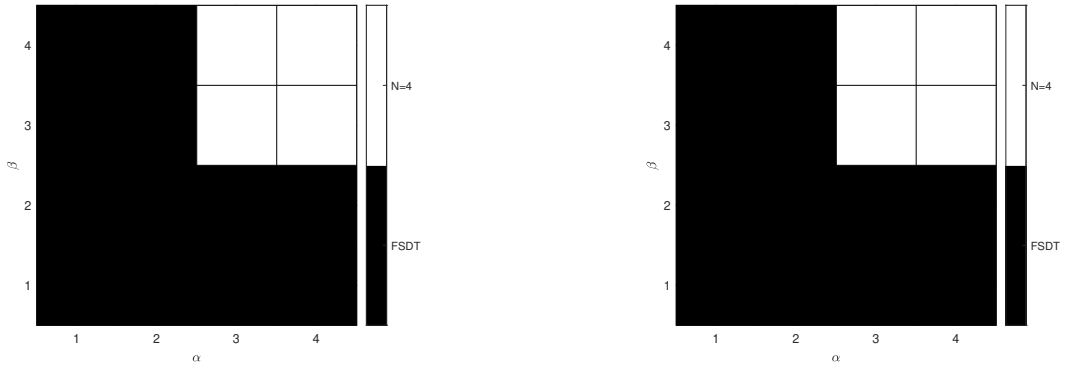


(b) NN, Err=0.618%.

Figura 6.39: Dodici N4, S-S, 0/90/0, a/h=100, R/a=5.

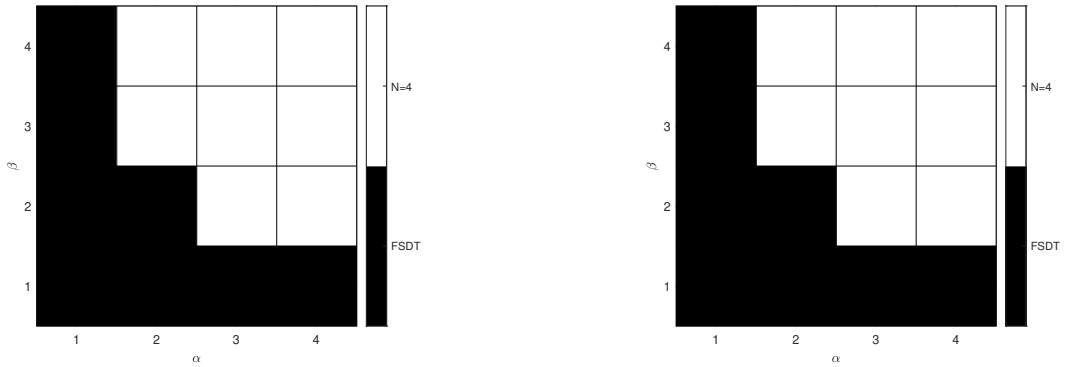
6.3.2 10 Frequenze

Nella riproduzione delle distribuzioni per le singole frequenze, la rete neurale fornisce nuovamente risultati ottimi, riuscendo a predire la corretta distribuzione in quasi tutte le casistiche. Ad ogni modo, nei casi in cui la distribuzione è differente, questa varia di un singolo elemento e produce un errore sul parametro di controllo che si differenzia con uno scostamento ridotto dal valore calcolato tramite software di calcolo.



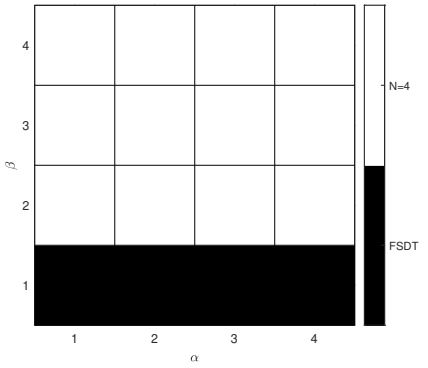
(a) FEM, Err=0.191%. (b) NN, Err=0.191%.

Figura 6.40: Frequenza I - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5.

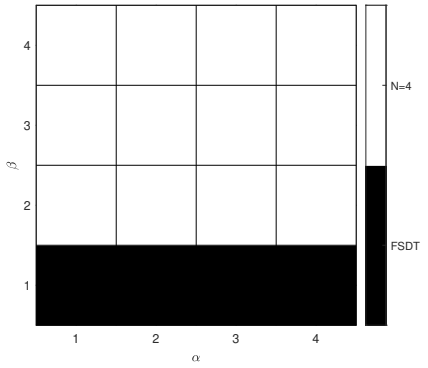


(a) FEM, Err=0.070%. (b) NN, Err=0.070%.

Figura 6.41: Frequenza I - Otto N4, S-S, 0/90/0, a/h=100, R/a=5.

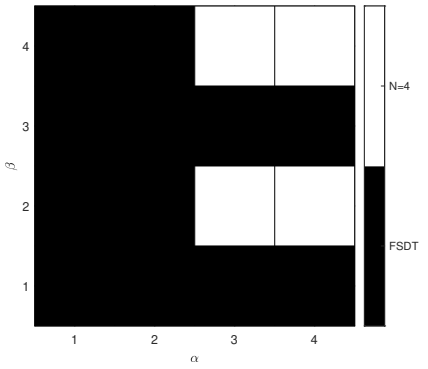


(a) FEM, Err=0.017%.

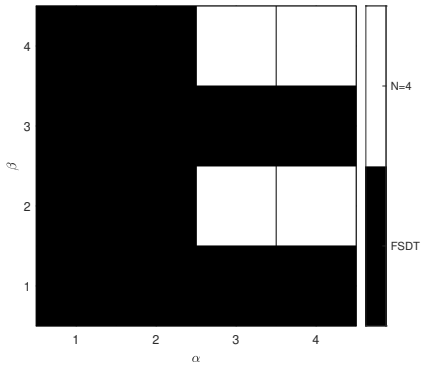


(b) NN, Err=0.017%.

Figura 6.42: Frequenza I - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5.

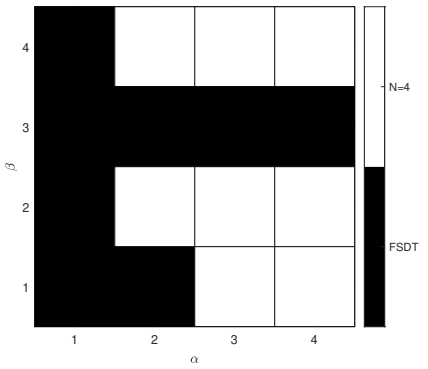


(a) FEM, Err=0.895%.

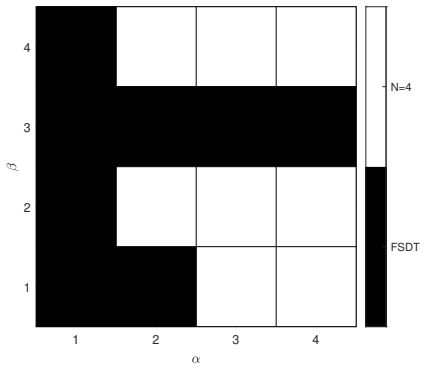


(b) NN, Err=0.900%.

Figura 6.43: Frequenza II - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5.

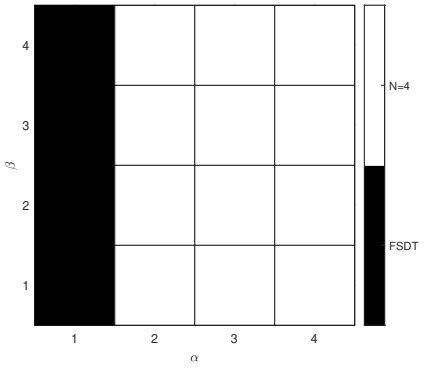


(a) FEM, Err=0.366%.

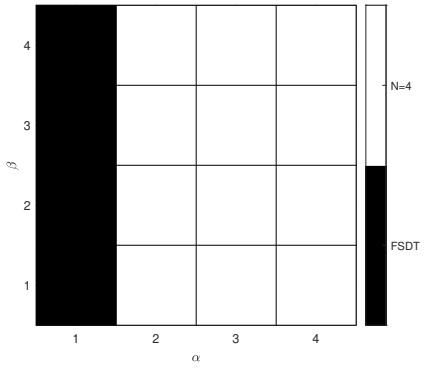


(b) NN, Err=0.376%.

Figura 6.44: Frequenza II - Otto N4, S-S, 0/90/0, a/h=100, R/a=5.

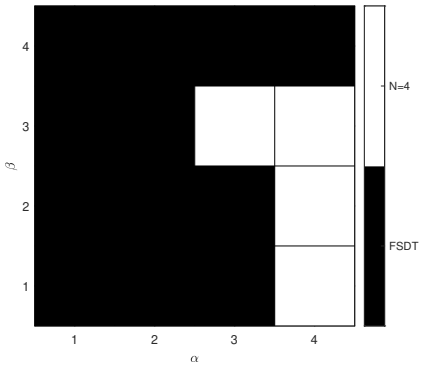


(a) FEM, Err=0.077%.

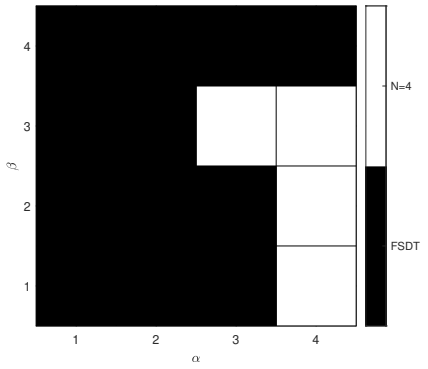


(b) NN, Err=0.075%.

Figura 6.45: Frequenza II - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5.

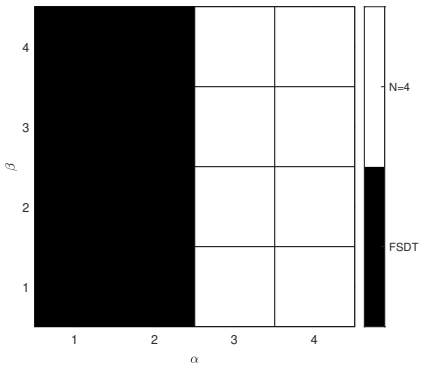


(a) FEM, Err=1.694%.

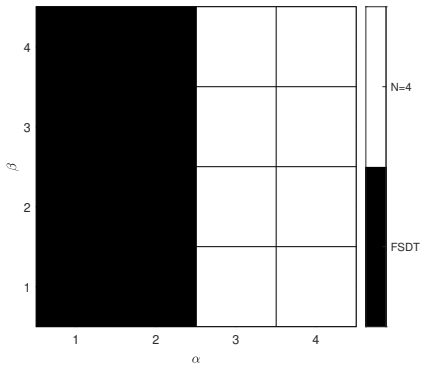


(b) NN, Err=1.708%.

Figura 6.46: Frequenza III - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5.

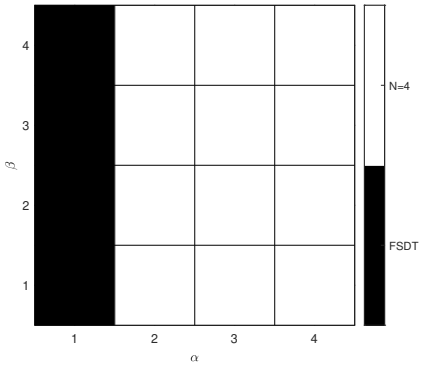


(a) FEM, Err=0.641%.

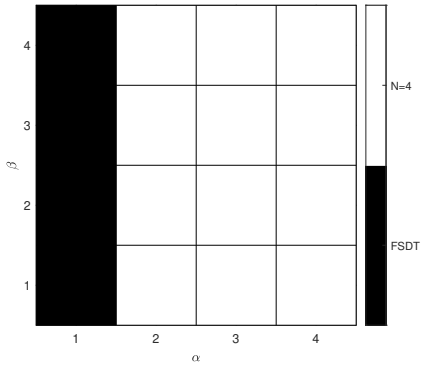


(b) NN, Err=0.642%.

Figura 6.47: Frequenza III - Otto N4, S-S, 0/90/0, a/h=100, R/a=5.

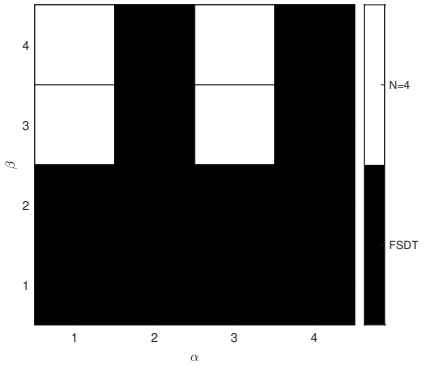


(a) FEM, Err=0.110%.

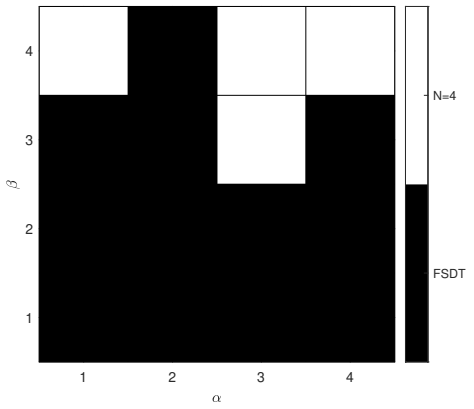


(b) NN, Err=0.111%.

Figura 6.48: Frequenza III - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5.

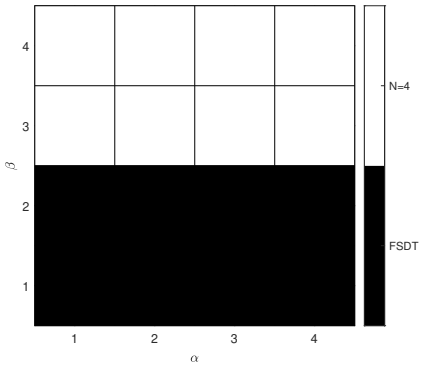


(a) FEM, Err=1.165%.

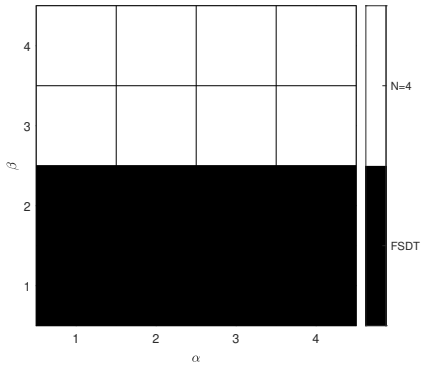


(b) NN, Err=1.170%.

Figura 6.49: Frequenza IV - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5.

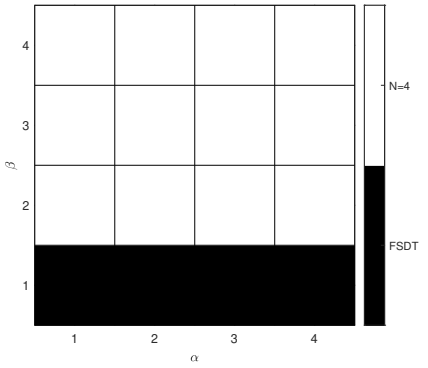


(a) FEM, Err=0.427%.

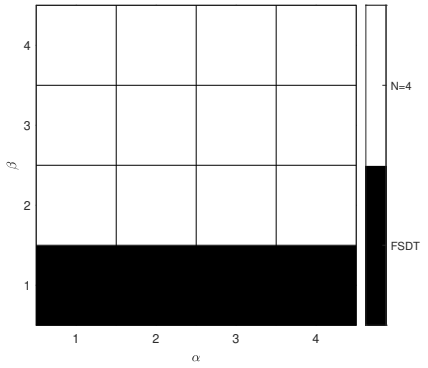


(b) NN, Err=0.427%.

Figura 6.50: Frequenza IV - Otto N4, S-S, 0/90/0, a/h=100, R/a=5.

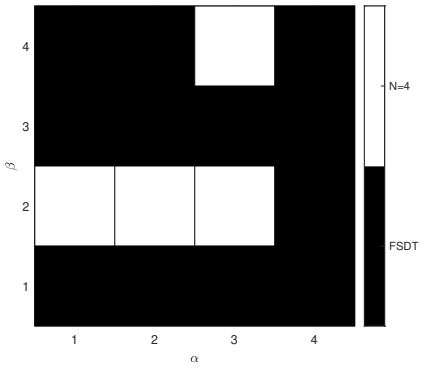


(a) FEM, Err=0.074%.

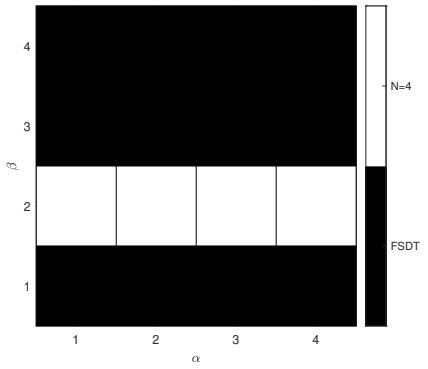


(b) NN, Err=0.068%.

Figura 6.51: Frequenza IV - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5.

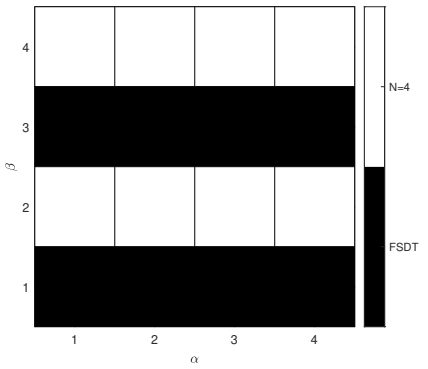


(a) FEM, Err=1.874%.

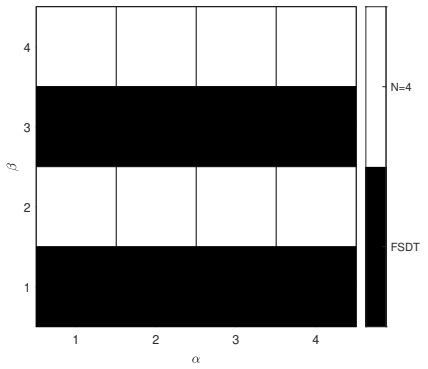


(b) NN, Err=1.857%.

Figura 6.52: Frequenza V - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5.

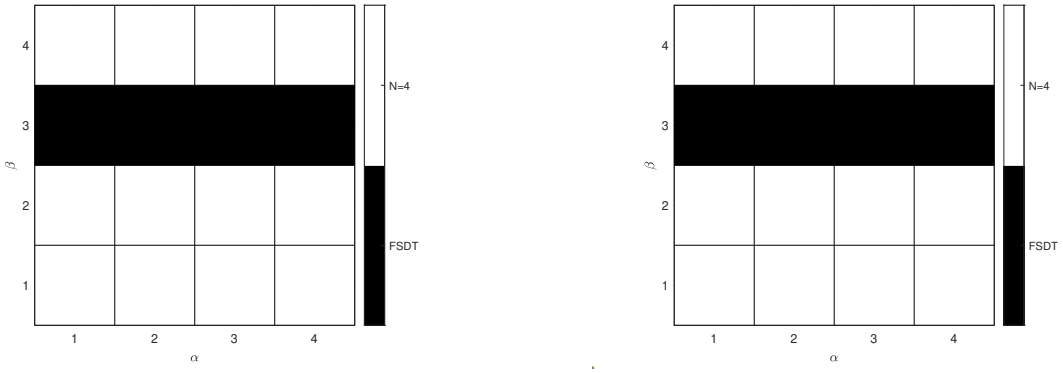


(a) FEM, Err=0.934%.



(b) NN, Err=0.936%.

Figura 6.53: Frequenza V - Otto N4, S-S, 0/90/0, a/h=100, R/a=5.



(a) FEM, Err=0.322%. (b) NN, Err=0.313%.

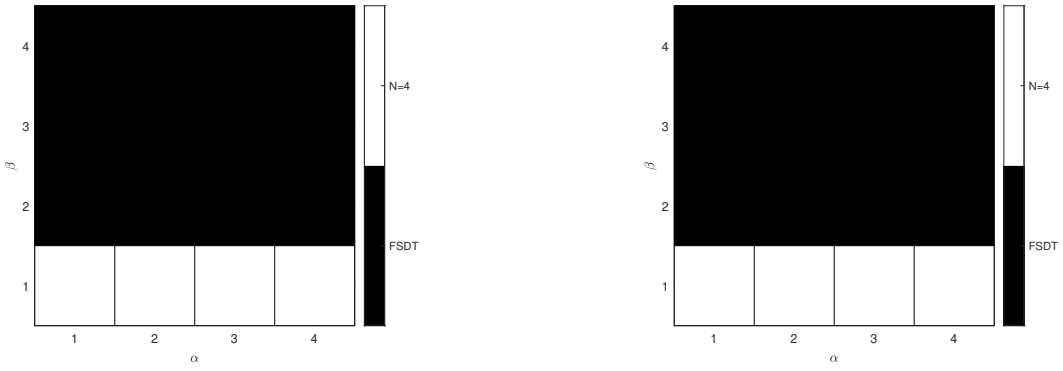
Figura 6.54: Frequenza V - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5.

6.4 C-F 0/90/0 a/h=10

In questa sezione si analizza la struttura mantenendo un rapporto di spessore pari ad $a/h = 10$, ma variando ora la condizione di vincolo con quella denominata clamped-free, che coincide con la condizione di incastro su due lati e libera sui due rimanenti.

6.4.1 Media

Nella tabella 6.5 si osserva che anche in queste condizioni di vincolo, la rete neurale riproduce con ottima accuratezza i valori degli errori per ogni numerosità di elementi definiti da una teoria del tipo N4. Questo conferma quanto ci si poteva aspettare, dato che la rete neurale artificiale funziona trovando dei legami matematici intrinseci e non dipende dalle condizioni geometriche e di vincolo della struttura. Inoltre,la rappresentazione delle distribuzioni di modelli di teorie non presenta differenze tra quella valutata tramite metodologia fem e quella predetta dalla rete neurale.

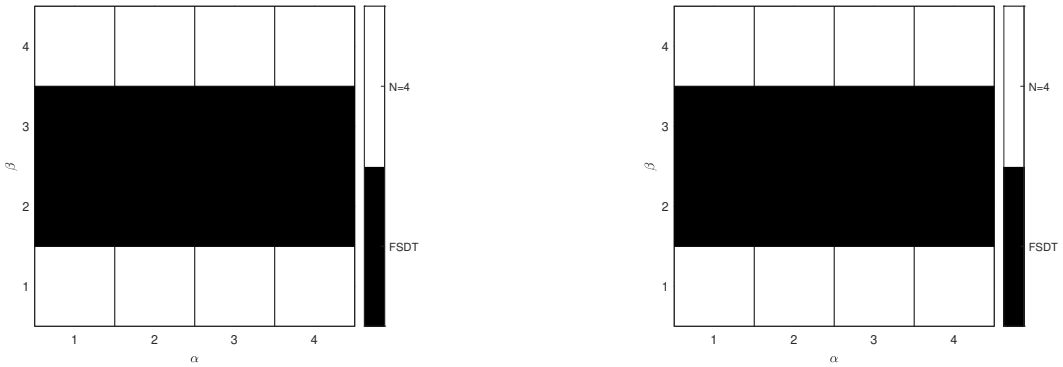


(a) FEM, Err=9.976%. (b) NN, Err=9.797%.

Figura 6.55: Quattro N4, C-F, 0/90/0, a/h=10, R/a=5.

Tabella 6.5: C-F, 0/90/0, a/h=10, R/a=5.

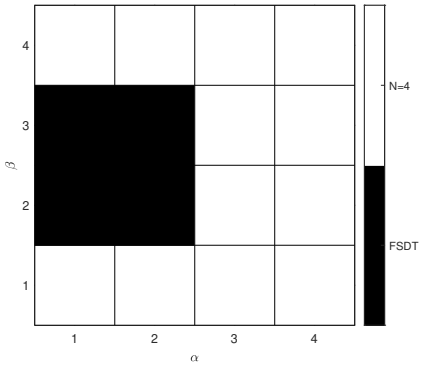
El. N4	Err% Fem	Err% NN
1	13.649	13.649
2	12.368	12.302
3	11.179	10.966
4	9.976	9.797
5	9.505	9.321
6	8.515	8.424
7	7.681	7.768
8	6.659	6.734
9	6.239	6.191
10	5.295	5.318
11	4.534	4.593
12	3.621	3.618
13	2.826	2.909
14	2.027	2.027
15	1.364	1.364



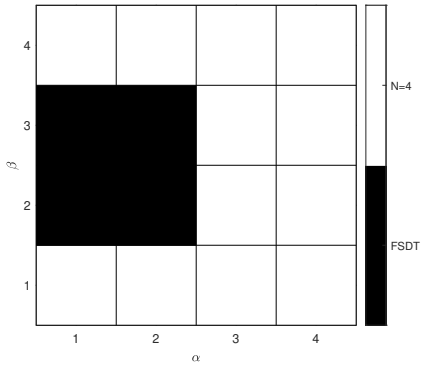
(a) FEM, Err=6.659%.

(b) NN, Err=6.734%.

Figura 6.56: Otto N4, C-F, 0/90/0, a/h=10, R/a=5.



(a) FEM, Err=3.621%.

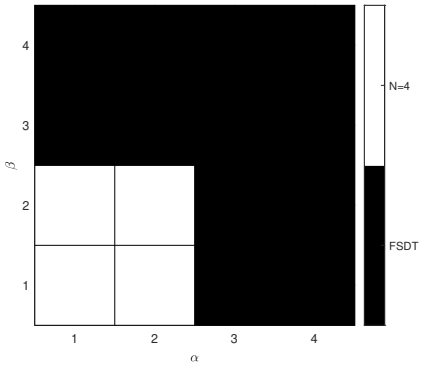


(b) NN, Err=3.618%.

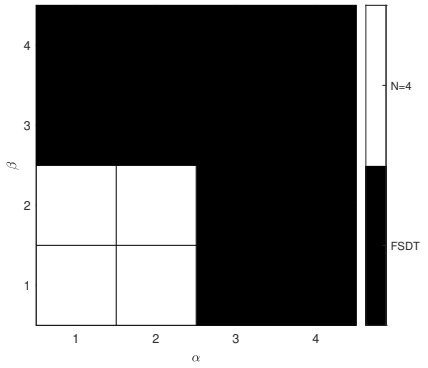
Figura 6.57: Dodici N4, C-F, 0/90/0, a/h=10, R/a=5.

6.4.2 10 Frequenze

In queste condizioni di vincolo, la rete neurale riproduce con alcune difficoltà le distribuzioni per le singole frequenze. Alcune risultano essere identiche alla controparte prodotta dall'analisi tramite il software strutturale, mentre altre presentano differenze talvolta più marcate rispetto ai casi osservati finora. Inoltre, quelle più differenti producono anche una prediction sull'errore percentuale il cui scostamento non si può sempre considerare trascurabile. Tale comportamento anomalo si può presuppore essere dovuto alla variabilità delle distribuzioni che si osservano tra le varie frequenze di questo sistema in esame.

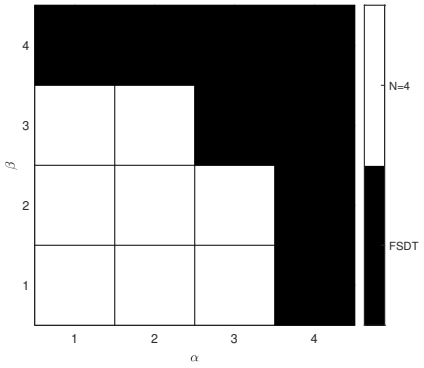


(a) FEM, Err=12.156%.

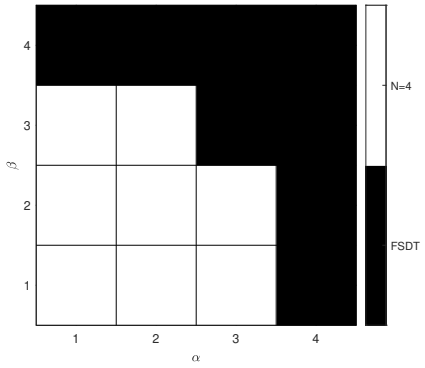


(b) NN, Err=12.146%.

Figura 6.58: Frequenza I - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5.

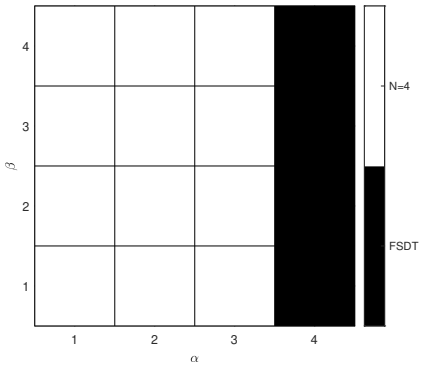


(a) FEM, Err=6.870%.

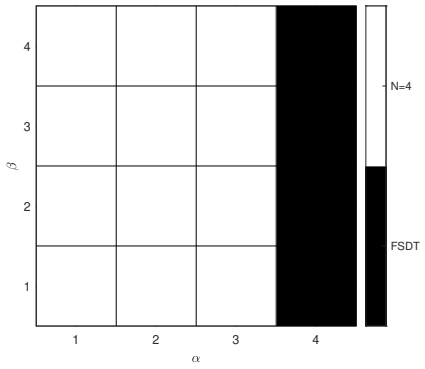


(b) NN, Err=6.936%.

Figura 6.59: Frequenza I - Otto N4, C-F, 0/90/0, a/h=10, R/a=5.

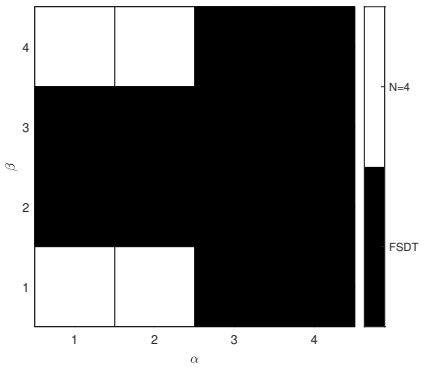


(a) FEM, Err=1.980%.

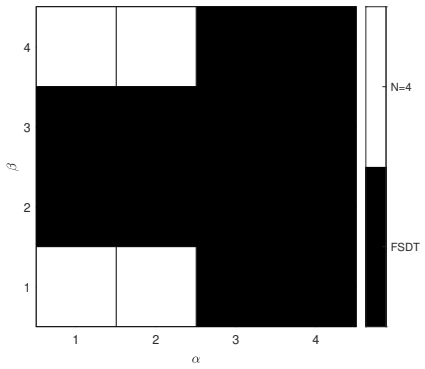


(b) NN, Err=1.896%.

Figura 6.60: Frequenza I - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5.

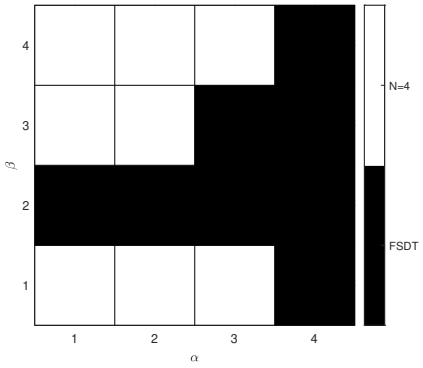


(a) FEM, Err=8.430%.

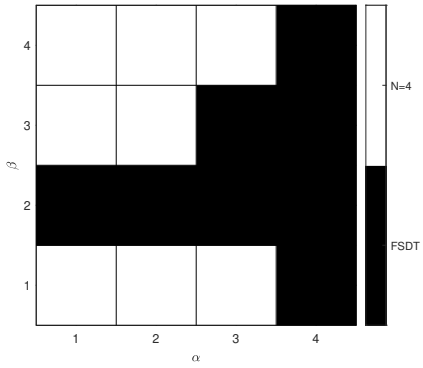


(b) NN, Err=8.428%.

Figura 6.61: Frequenza II - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5.

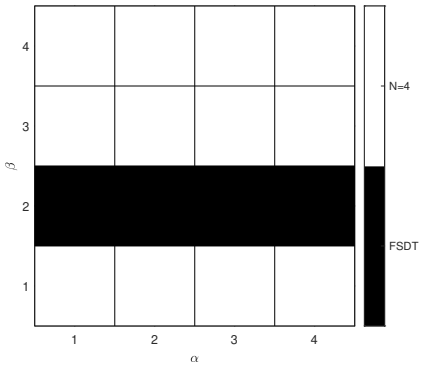


(a) FEM, Err=3.763%.

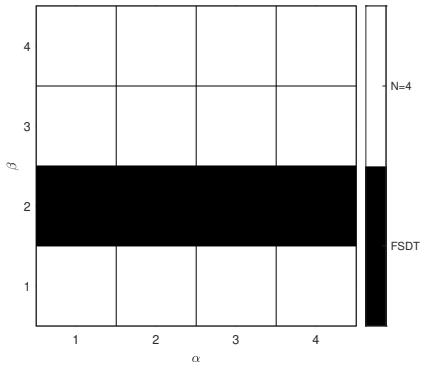


(b) NN, Err=3.691%.

Figura 6.62: Frequenza II - Otto N4, C-F, 0/90/0, a/h=10, R/a=5.

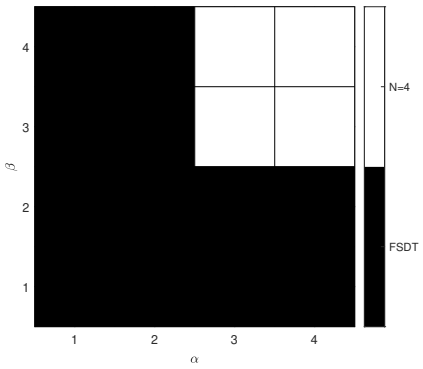


(a) FEM, Err=1.378%.

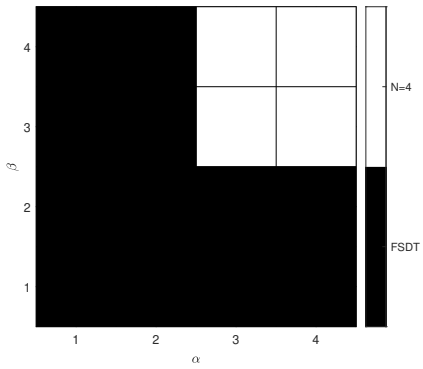


(b) NN, Err=1.378%.

Figura 6.63: Frequenza II - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5.

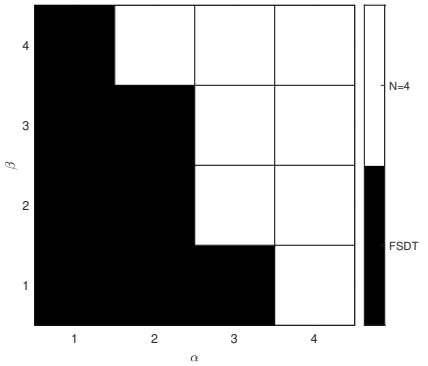


(a) FEM, Err=6.197%.

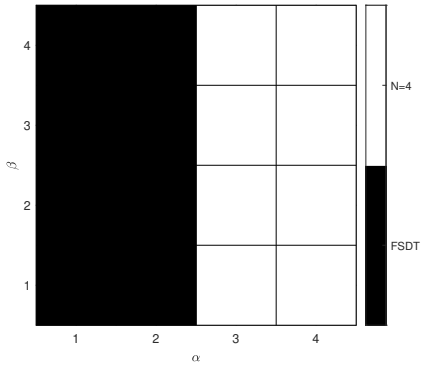


(b) NN, Err=6.188%.

Figura 6.64: Frequenza III - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5.

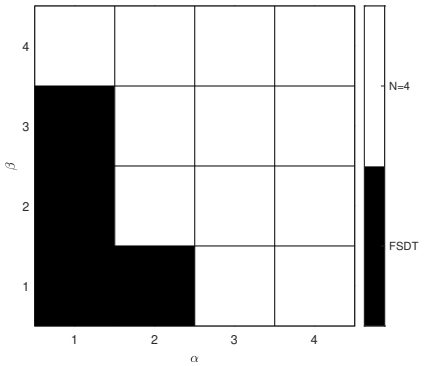


(a) FEM, Err=4.060%.

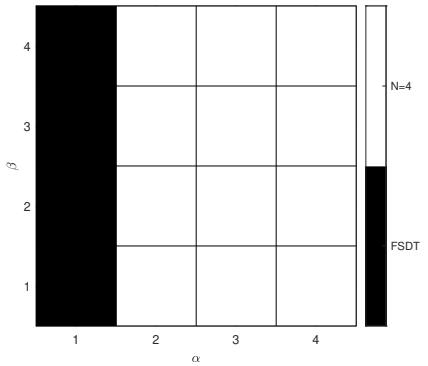


(b) NN, Err=3.996%.

Figura 6.65: Frequenza III - Otto N4, C-F, 0/90/0, a/h=10, R/a=5.

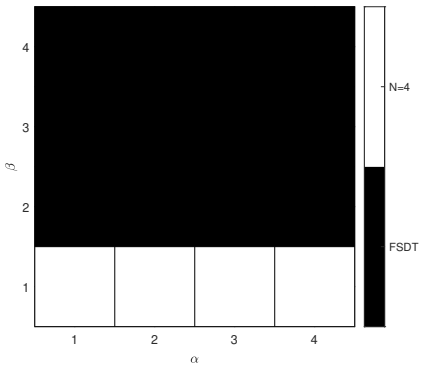


(a) FEM, Err=1.910%.

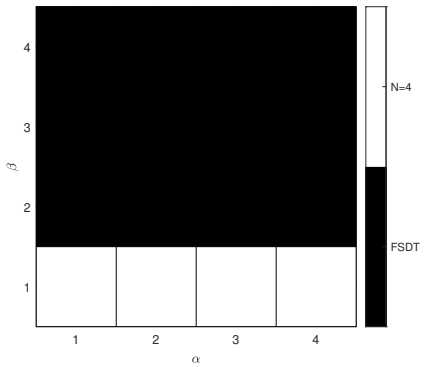


(b) NN, Err=1.743%.

Figura 6.66: Frequenza III - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5.

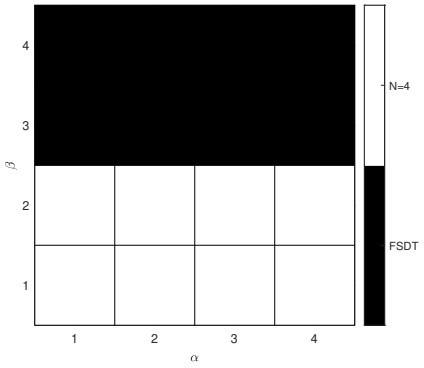


(a) FEM, Err=8.836%.

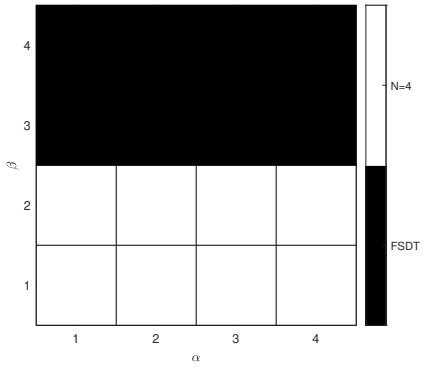


(b) NN, Err=8.927%.

Figura 6.67: Frequenza IV - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5.

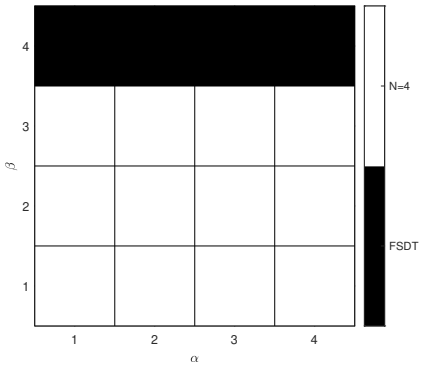


(a) FEM, Err=3.712%.

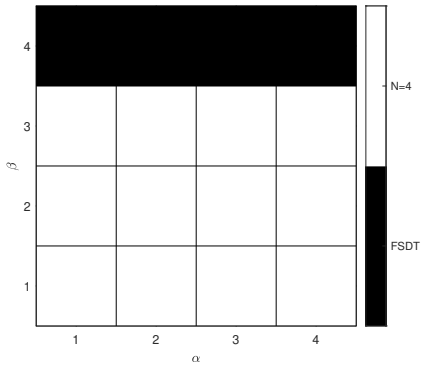


(b) NN, Err=3.882%.

Figura 6.68: Frequenza IV - Otto N4, C-F, 0/90/0, a/h=10, R/a=5.

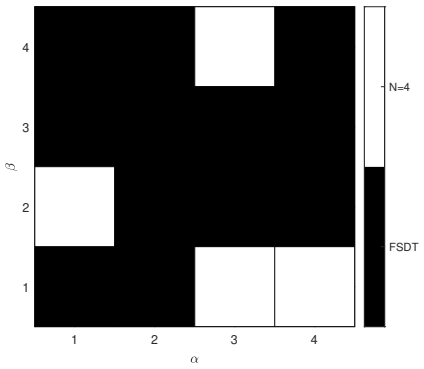


(a) FEM, Err=2.035%.

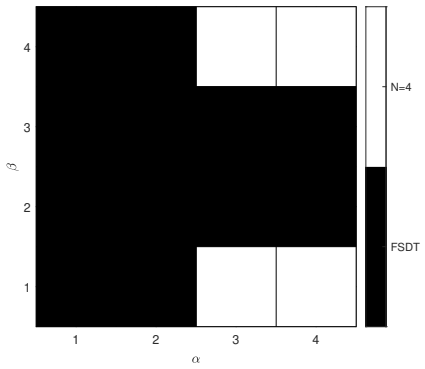


(b) NN, Err=2.002%.

Figura 6.69: Frequenza IV - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5.

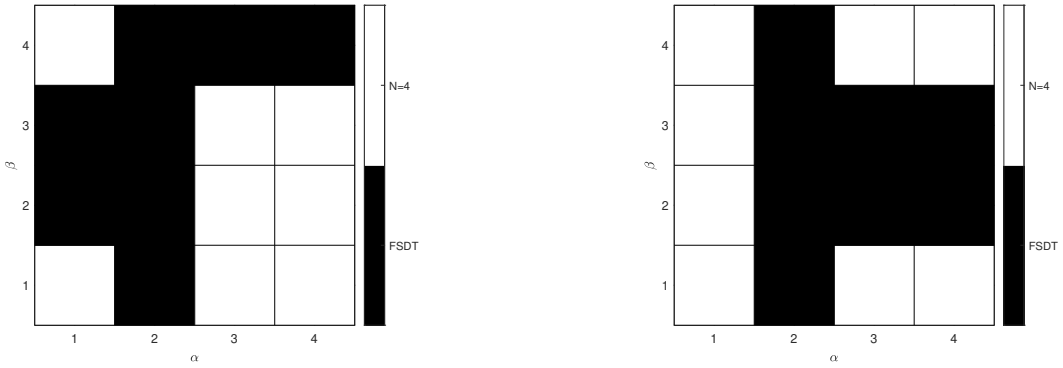


(a) FEM, Err=7.470%.



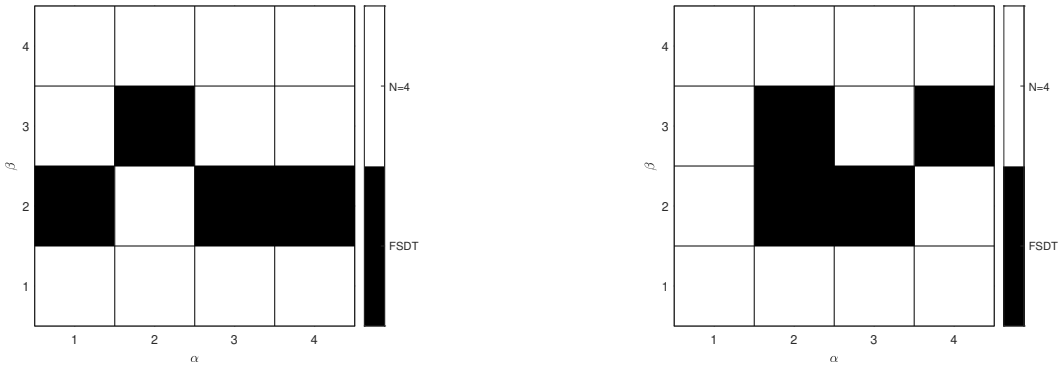
(b) NN, Err=9.491%.

Figura 6.70: Frequenza V - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5.



(a) FEM, Err=1.938%. (b) NN, Err=3.961%.

Figura 6.71: Frequenza V - Otto N4, C-F, 0/90/0, a/h=10, R/a=5.



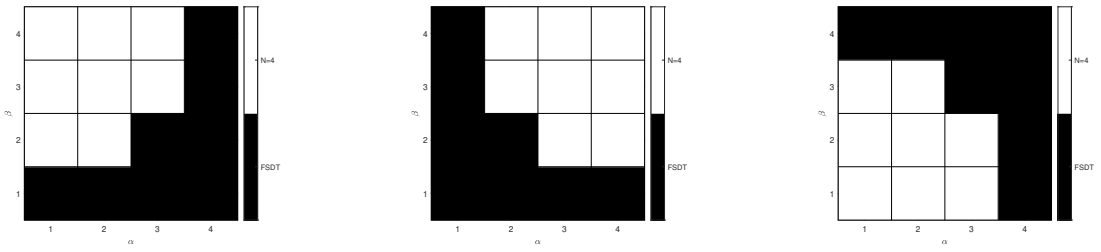
(a) FEM, Err=0.054%. (b) NN, Err=0.545%.

Figura 6.72: Frequenza V - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5.

6.5 Riepilogo

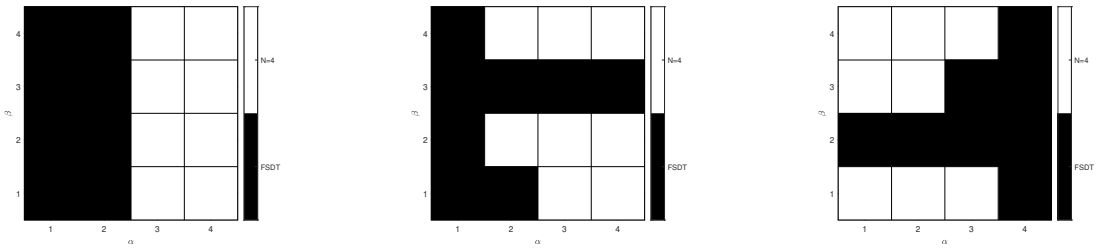
In quest'ultima sezione del capitolo viene proposto un riepilogo dei risultati ottenuti dall'analisi global/local tramite l'utilizzo della rete neurale artificiale. Sono presentate le distribuzioni ottimali di teorie nella casistica di otto elementi di tipo N4 ed otto elementi di tipo FDST, confrontate ad ogni frequenza tra i tre sistemi strutturali analizzati. Analizzando quanto riportato, risulta immediato osservare come le tre casistiche comportino distribuzioni sempre differenti tra loro, all'interno delle medesime frequenze. Inoltre, solo talvolta in certe frequenze si ripetono gli stessi pattern di utilizzo di teorie strutturali di ordine elevato. La grande variabilità che si nota, comporta dunque una prima osservazione su quanto sia necessario uno studio global/local per raffinare i risultati ottenibili ed una seconda osservazione su quanto questo vada ad accrescere il costo computazionale ed il tempo di analisi richiesti.
L'utilizzo della rete neurale proposto in questo capitolo comporta dei grandi benefici,

in quanto l'accuratezza si è dimostrata essere rilevante, soprattutto considerando il costo computazionale ridotto. Ogni singola analisi global/local effettuata tramite il software di calcolo strutturale infatti necessitava di un minimo di sei ore prima di fornire i risultati. Le rete neurale, impiegava un tempo di circa trenta minuti comprensivo di fase di training e fase di prediction. In tal modo sarebbe possibile ottenere le informazioni sulle distribuzioni di teoria ottimali per una qualsiasi frequenza in un tempo notevolmente ridotto.



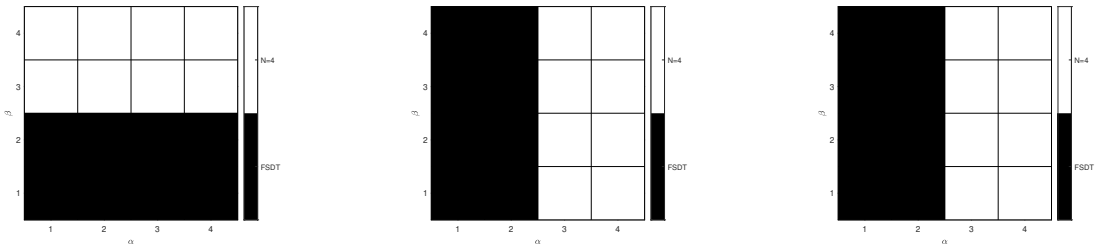
(a) S-S, a/h=10. (b) S-S, s/h=100. (c) C-F, a/h=10.

Figura 6.73: Frequenza I.



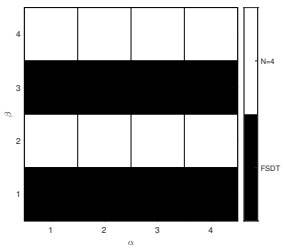
(a) S-S, a/h=10. (b) S-S, s/h=100. (c) C-F, a/h=10.

Figura 6.74: Frequenza II.

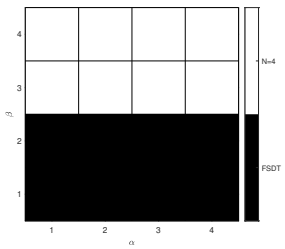


(a) S-S, a/h=10. (b) S-S, s/h=100. (c) C-F, a/h=10.

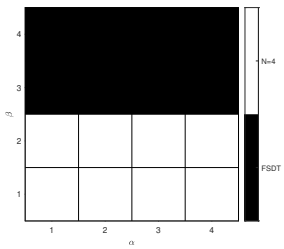
Figura 6.75: Frequenza III.



(a) S-S, a/h=10.

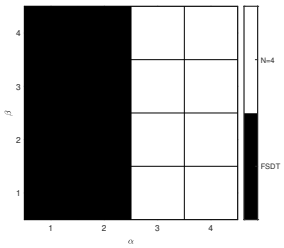


(b) S-S, s/h=100.

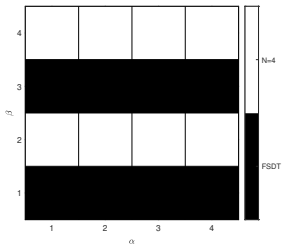


(c) C-F, a/h=10.

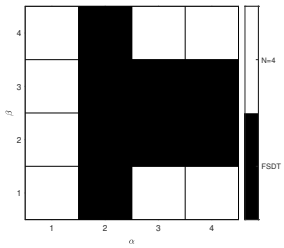
Figura 6.76: Frequenza IV.



(a) S-S, a/h=10.

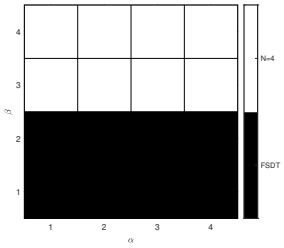


(b) S-S, s/h=100.

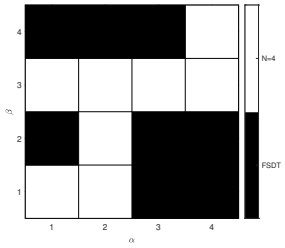


(c) C-F, a/h=10.

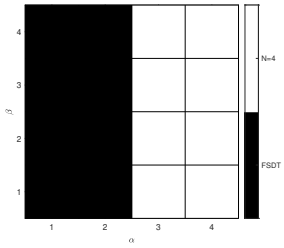
Figura 6.77: Frequenza V.



(a) S-S, a/h=10.

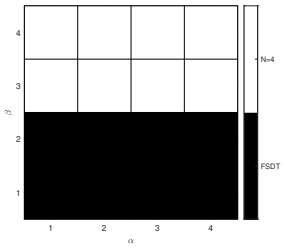


(b) S-S, s/h=100.

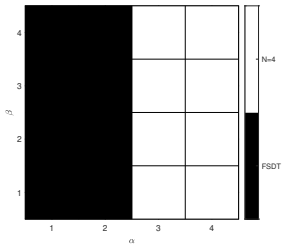


(c) C-F, a/h=10.

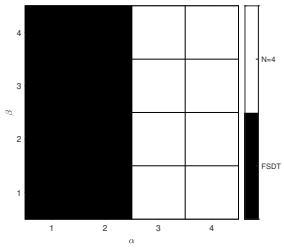
Figura 6.78: Frequenza VI.



(a) S-S, a/h=10.

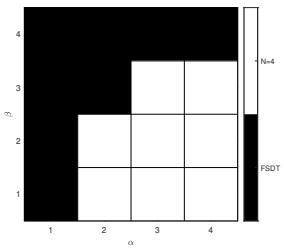


(b) S-S, s/h=100.

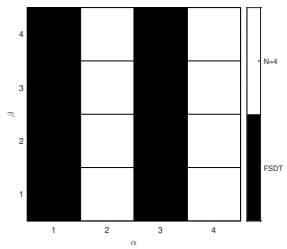


(c) C-F, a/h=10.

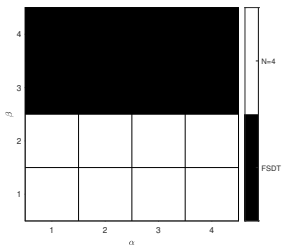
Figura 6.79: Frequenza VII.



(a) S-S, a/h=10.

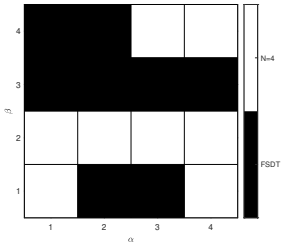


(b) S-S, s/h=100.

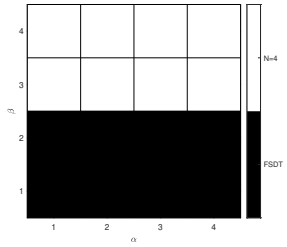


(c) C-F, a/h=10.

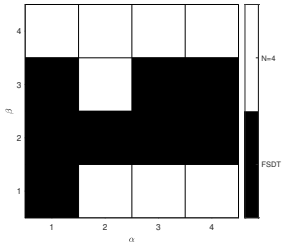
Figura 6.80: Frequenza VIII.



(a) S-S, a/h=10.

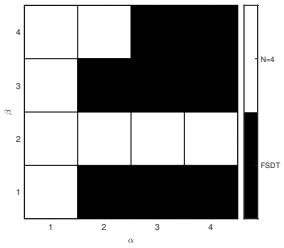


(b) S-S, s/h=100.

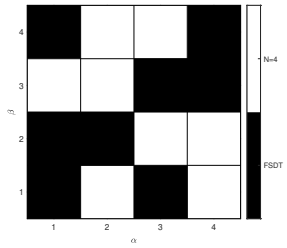


(c) C-F, a/h=10.

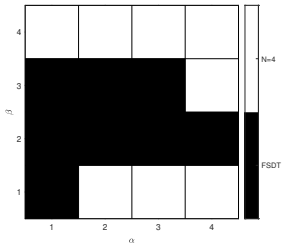
Figura 6.81: Frequenza IX.



(a) S-S, a/h=10.



(b) S-S, s/h=100.



(c) C-F, a/h=10.

Figura 6.82: Frequenza X.

Capitolo 7

Analisi Failure index

Un ulteriore aspetto molto importante nell'ambito delle analisi strutturali, in particolare quando è previsto l'utilizzo di materiali compositi, è la valutazione dei failure index in differenti punti della struttura, a seconda dello stato di carico e di tensione previsti. Questo tipo di analisi permette di identificare i punti critici di una struttura in cui è più probabile il verificarsi di malfunzionamenti di vari tipi, andando così ad agire opportunamente per evitare che ciò possa accadere per poter garantire la massima sicurezza ed affidabilità.
Attualmente esistono molteplici criteri per la determinazione degli indici di failure, ognuno con la propria applicazione opportuna a seconda del sistema in considerazione e del tipo di rottura prevista. Se ne può trovare un confronto dettagliato in [30].
In questo lavoro di tesi si è scelto di considerare il criterio di Hashin esposto in [31], in cui vengono considerati diverse combinazioni di termini per distinguere tra modalità di failure causate da compressione o trazione, nei casi di fibra e matrice di materiali compositi.
A seguire sono riportate le disuguaglianze adottate per il calcolo dei failure index secondo il criterio di Hashin.

Fibra a compressione FC ($\sigma_{11} < 0$):

$$-\frac{\sigma_{11}}{X^c} \geq 1 \tag{7.1}$$

Fibra a tensione FT ($\sigma_{11} \geq 0$):

$$\left(\frac{\sigma_{11}}{X^T}\right)^2 + \left(\frac{\tau_{12}}{S^L}\right)^2 \geq 1 \tag{7.2}$$

Matrice a compressione MC ($\sigma_{22} \geq 0$):

$$\left(\frac{\sigma_{22}}{Y^T}\right)^2 + \left(\frac{\tau_{12}}{S^L}\right)^2 \geq 1 \tag{7.3}$$

Matrice a tensione MT ($\sigma_{22} < 0$):

$$\left(\frac{\sigma_{22}}{2S^T}\right)^2 + \left[\left(\frac{Y^C}{2S^T}\right) - 1\right]\frac{\sigma_{22}}{Y^C} + \left(\frac{\tau_{12}}{S^L}\right)^2 \geq 1 \tag{7.4}$$

Si vuole dunque procedere con l'implementazione di questo tipo di analisi strutturale all'interno della rete neurale artificiale sviluppata. Il processo di implementazione coincide con la metodologia già illustrata precedentemente in questo lavoro di tesi, ma con le opportune attenzioni a riguardo dei dati di output ottenuti dal software di calcolo strutturale che andranno ad essere utilizzati come input per la rete neurale. Per questo tipo di analisi, il software di calcolo produce in output le 2^{12} combinazioni di termini attivi (utilizzando sempre una funzione di espansione di Taylor di grado massimo quattro e considerando i termini di ordine zero sempre presenti), per ognuno dei punti richiesti esplicitamente. Indicando dunque con n il numero di punti, si avranno 2^{12} combinazioni per ognuno dei punti in esame, ottenendo così $n \times 2^{12}$ risultati per cui verranno calcolati i quattro failure index di Hashin di cui sono riportate le definizioni in precedenza. Si fa notare che ad ogni combinazione saranno presenti quattro valori di failure index, ma due di questi saranno nulli, in quanto il codice di calcolo fornisce solo il valore di quello maggiore tra compressione e trazione, per ciascuno dei due casi matrice e fibra.
I dati di input della rete neurale in questa applicazione diventano quindi le combinazioni di termini attivi ed inattivi della teoria, oltre alle coordinate dei punti di cui si vuole valutare i failure index. Il parametro di controllo su cui andrà ad operare è l'errore percentuale relativo a ciascuno dei failure index. Questo tipo di operazione matematica crea all'interno dello script Python dei valori di tipo *NaN* ed *inf* dovuti all'utilizzo dello zero (relativo ai failure index nulli) nelle divisioni per il calcolo dell'errore. La rete neurale artificiale non può funzionare con questo tipo di valori, pertanto nei casi in cui erano presenti, si è impostanto arbitrariamente il valore -1, in modo tale da essere ben riconoscibile durante la fase di analisi dei risultati. Le fasi di training e prediction della rete neurale avvengono dunque nella riproduzione degli errori relativi ai quattro failure index di ciascun punto in esame. I risultati riportati nell'elaborato sono forniti con un unico valore corrispondente alla media dei due failure index più rilevanti di ogni punto.
Viene utilizzata la stessa struttura di tipo guscio dei capitoli precedenti con le medesime caratteristiche geometriche $a = b$, $R_\alpha = R_\beta = R$, rapporto $R/a = 5$. Il materiale considerato ha proprietà $E_1/E_2 = 25$, $G_{12}/E_2 = G_{13}/E_2 = 0.5$, $G_{13}/E_2 = 0.2$, $\nu = 0.25$. Le condizioni di vincolo sono quelle del semplice appoggio ed è stato applicato un carico con andamento bi-sinusoidale di intensità massima al centro della piastra pari ad 1 MPa. Vengono sfruttate le opportune condizoni di simmetria, in modo da effettuare le analisi solamente su una porzione della struttura pari ad un quarto del guscio.
Sono stati scelti arbitrariamente quattro punti appartenenti alla struttura, scelti a coppie sulla faccia superiore ed inferiore del guscio. Le coordinate dei punti sono riportate nelle tabelle 7.1 e 7.2. Le analisi sono state poi eseguite variando il rapporto di spessore a/h.

Tabella 7.1: Coordinate dei punti. a/h=10.

ID	x	y	z
1	0	0	0.05
2	0	0	-0.05
3	0.5	0.5	0.05
4	0.5	0.5	-0.05

Tabella 7.2: Coordinate dei punti. a/h=100.

ID	x	y	z
1	0	0	0.005
2	0	0	-0.005
3	0.5	0.5	0.005
4	0.5	0.5	-0.005

7.1 Output multiplo

Nelle tabelle 7.3 e 7.4 sono riportati i valori ottenuti come media degli errori percentuali sui failure index, calcolati rispettivamente tramite il software di analisi strutturale e tramite la rete neurale artificiale, sulla struttura con rapporto $a/h = 10$. Analogamente, nelle tabelle 7.5 e 7.6 sono riportati i risultati relativi alla struttura con rapporto $a/h = 100$.
Si può notare che in entrambe le casistiche la rete neurale non fornisce risultati affidabili, variando talvolta di diversi ordini di grandezza, o fornendo valori nulli laddove non dovrebbero esserlo. Successivamente sono state svolte diverse analisi sugli stessi punti della struttura, provando a variare alcuni parametri della rete quali loss function e funzioni di attivazione dei vari layer, ma non hanno portato miglioramenti rispetto i risultati già riportati nelle tabelle. La motivazione più plausibile di questo problema può risiedere nel fatto che i dati di input sono stati manipolati arbitrariamente per sopperire a dei valori incongruenti e questo arteficio potrebbe aver portato la rete neurale a non essere in grado di stabilire le giuste relazioni matematiche intrinseche all'analisi.
A seguito dell'aver confermato la scarsità nella riproduzione dei risultati con questo metodo, viene presentato un nuovo approccio sperimentato nella sezione a seguire.

Tabella 7.3: Semplice appoggio, 0/90/0, R/a=5, a/h=10 - Valori analisi statica.

DOF	E_1	E_2	E_3	E_4
3	96.157	100	99.997	100
4	95.859	92.437	93.945	85.751
5	25.103	21.570	19.935	10.289
6	4.816	1.962	1.985	9.378
7	2.620	0.661	2.095	1.064
8	0.820	0.309	0.424	0.199
9	0.164	0.312	0.267	0.153
10	1.830×10^{-2}	0.319	0.126	0.156
11	1.132×10^{-2}	0.294	1.608×10^{-2}	0.164
12	7.496×10^{-4}	0.222	1.015×10^{-2}	9.053×10^{-2}
13	6.574×10^{-3}	0.103	1.304×10^{-2}	4.694×10^{-2}
14	8.381×10^{-3}	2.317×10^{-2}	8.373×10^{-3}	6.710×10^{-3}
15	0	0	0	0

Tabella 7.4: Semplice appoggio, 0/90/0, R/a=5, a/h=10 - Valori rete neurale.

DOF	E_1	E_2	E_3	E_4
3	99.952	99.709	99.757	99.722
4	98.888	92.612	93.784	85.820
5	23.470	17.287	21.717	14.090
6	5.548	0.797	4.781	6.404
7	2.361	0.614	3.065	1.721
8	1.127	0.341	1.027	0.625
9	0.311	0.236	0.296	0.291
10	4.389×10^{-2}	0.291	0.110	0.135
11	0.128	0.410	3.765×10^{-2}	0.110
12	3.987×10^{-2}	0.603	1.041×10^{-2}	0.119
13	0	0.107	4.293×10^{-2}	0.157
14	1.268×10^{-2}	2.506×10^{-2}	4.536×10^{-3}	3.558×10^{-3}
15	1.238×10^{-2}	4.458×10^{-3}	5.042×10^{-3}	3.606×10^{-3}

Tabella 7.5: Semplice appoggio, 0/90/0, R/a=5, a/h=100 - Valori analisi statica.

DOF	E_1	E_2	E_3	E_4
3	99.961	100	100	100
4	99.851	98.975	87.849	84.131
5	0.597	21.894	26.092	13.555
6	0.221	11.732	21.535	11.223
7	7.808×10^{-2}	8.186	9.223	10.905
8	1.942×10^{-2}	4.629	4.848	4.966
9	4.268×10^{-3}	3.854	8.040×10^{-3}	2.513×10^{-2}
10	2.764×10^{-4}	1.905	1.631×10^{-3}	1.025×10^{-2}
11	4.762×10^{-4}	1.642	1.151×10^{-3}	4.334×10^{-3}
12	1.090×10^{-3}	0.577	1.154×10^{-3}	2.024×10^{-3}
13	8.367×10^{-4}	0.198	5.676×10^{-4}	6.037×10^{-4}
14	2.481×10^{-5}	2.116×10^{-3}	1.259×10^{-5}	1.740×10^{-5}
15	0	0	0	0

Tabella 7.6: Semplice appoggio, 0/90/0, R/a=5, a/h=100 - Valori rete neurale.

DOF	E_1	E_2	E_3	E_4
3	99.945	99.899	99.907	99.914
4	99.829	99.007	87.928	84.095
5	0.610	21.914	26.000	14.143
6	0.186	11.725	21.931	11.723
7	1.713×10^{-2}	9.204	9.370	10.745
8	0	4.699	5.160	5.835
9	0	2.627	0.101	0.394
10	0	2.149	0	0
11	0	0.952	0	0
12	0	0.297	0	0
13	0	0.189	0	0
14	4.291×10^{-4}	1.152×10^{-2}	0	9.942×10^{-3}
15	5.876×10^{-2}	1.685×10^{-2}	1.678×10^{-2}	0

7.2 Output singolo

Nella sezione precedente si è osservato che la rete neurale non era in grado di riprodurre fedelmente i valori riguardanti l'insieme dei quattro failure index considerati. Per questo motivo si è deciso di testare un nuovo approccio, andando a prendere in considerazione un singolo e medesimo failure index per ciascuno dei punti considerati.

La struttura considerata è sempre il guscio con le caratteristiche già riportate, il rapporto di spessore utilizzato è $a/h = 100$. Il procedimento viene svolto considerando come parametro di controllo l'errore sul failure index di Hashin per la matrice a compressione MC, ma in maniera analoga può essere svolto per gli altri indici. In tabella 7.7 sono riportate le coordinate dei punti scelti per questa analisi. Ne sono stati scelti solamente tre, per alleggerire il carico computazionale in questa fase di test. Inoltre sono stati scelti opportunamente in maniera tale che il failure index fosse non nullo in due di questi punti e nullo nel terzo (dovuto al fatto che risulta maggiore il rispettivo failure index in trazione, pertanto il software strutturale riporta lo zero come valore in compressione).

Tabella 7.7: Coordinate dei punti. a/h=100.

ID	x	y	z
1	0.25	0.25	-0.005
2	0.5	0.5	0.005
3	0.5	0.5	-0.005

Nelle tabelle 7.8 e 7.9 sono riportati i risultati più accurati ottenuti per questa casistica. In particolare, è stata variata la loss function con cui operava la rete neurale artificiale, andando ad utilizzare la *MeanSquaredError*, capace di produrre prediction migliori in questo tipo di analisi.
Effettuando un confronto tra codice di calcolo strutturale e rete neurale artificiale, si nota che l'accuratezza è migliorata rispetto ai casi della sezione precedente, anche se può considerarsi valida solo per alcune delle numerosità di termini attivi. In particolare, nelle numerosità più elevate, si ottiene un errore relativo al failure index sempre più ridotto, però la rete neurale sembra non riuscire ad identificarlo con correttezza al di sotto di un certo limite. A seguire, sono riportate le distribuzioni di termini attivi per i BTD dei punti 1 e 3, presentando il confronto tra codice di calcolo e rete neurale.

Tabella 7.8: Semplice appoggio, 0/90/0, R/a=5, a/h=100 - Valori analisi statica.

DOF	E_1	E_2	E_3
3	100	0	100
4	85.811	0	86.219
5	16.627	0	16.456
6	15.967	0	15.785
7	9.780	0	9.916
8	5.513	0	5.592
9	3.275×10^{-2}	0	4.144×10^{-2}
10	1.207×10^{-2}	0	1.245×10^{-2}
11	1.322×10^{-3}	0	3.721×10^{-4}
12	1.068×10^{-3}	0	1.425×10^{-5}
13	2.708×10^{-4}	0	1.768×10^{-6}
14	2.414×10^{-5}	0	2.784×10^{-5}
15	0	0	0

Tabella 7.9: Semplice appoggio, 0/90/0, R/a=5, a/h=100 - Valori rete neurale.

DOF	E_1	E_2	E_3
3	99.995	1.832×10^{-8}	99.995
4	85.813	9.744×10^{-9}	86.215
5	17.117	4.423×10^{-9}	16.902
6	16.159	3.488×10^{-9}	16.013
7	10.083	2.266×10^{-9}	10.210
8	5.428	1.582×10^{-9}	5.548
9	4.217×10^{-2}	1.994×10^{-9}	4.820×10^{-2}
10	3.656×10^{-2}	2.964×10^{-9}	4.164×10^{-2}
11	3.506×10^{-2}	4.249×10^{-9}	4.019×10^{-2}
12	3.451×10^{-2}	1.528×10^{-8}	3.955×10^{-2}
13	3.478×10^{-2}	6.390×10^{-8}	3.983×10^{-2}
14	3.501×10^{-2}	3.319×10^{-7}	4.039×10^{-2}
15	3.666×10^{-2}	1.616×10^{-4}	4.223×10^{-2}

Tabella 7.10: BTD Punto 1 (0.25, 0.25, -0.005).

DOF	$u_{\alpha1}$	$u_{\beta1}$	u_{z1}	$u_{\alpha2}$	$u_{\beta2}$	u_{z2}	$u_{\alpha3}$	$u_{\beta3}$	u_{z3}	$u_{\alpha4}$	$u_{\beta4}$	u_{z4}	$u_{\alpha5}$	$u_{\beta5}$	u_{z5}
14 FEM	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲
14 NN	▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲
13 FEM	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲	△	▲	▲
13 NN	▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	△	▲
12 FEM	▲	▲	▲	▲	▲	▲	▲	△	▲	△	▲	▲	▲	△	▲
12 NN	▲	▲	▲	▲	▲	▲	△	△	▲	▲	△	▲	▲	▲	▲
11 FEM	▲	▲	▲	▲	▲	▲	△	△	▲	△	▲	▲	△	▲	▲
11 NN	▲	▲	▲	▲	▲	▲	△	▲	▲	△	△	▲	▲	△	▲
10 FEM	▲	▲	▲	▲	▲	▲	△	▲	▲	△	△	▲	△	△	▲
10 NN	▲	▲	▲	▲	▲	▲	△	△	▲	△	△	▲	▲	△	▲
9 FEM	▲	▲	▲	▲	▲	▲	△	△	▲	△	△	▲	△	△	▲
9 NN	▲	▲	▲	▲	▲	▲	△	△	▲	△	△	▲	△	△	▲
8 FEM	▲	▲	▲	▲	▲	△	△	△	▲	△	△	▲	△	△	▲
8 NN	▲	▲	▲	▲	▲	△	△	△	▲	△	△	▲	△	△	▲
7 FEM	▲	▲	▲	▲	▲	△	△	△	▲	△	△	△	△	△	▲
7 NN	▲	▲	▲	▲	▲	△	△	△	▲	△	△	△	△	△	▲
6 FEM	▲	▲	▲	△	▲	△	△	△	▲	△	△	△	△	△	▲
6 NN	▲	▲	▲	△	▲	△	△	△	▲	△	△	△	△	△	▲
5 FEM	▲	▲	▲	△	△	△	△	△	▲	△	△	△	△	△	▲
5 NN	▲	▲	▲	△	△	△	△	△	▲	△	△	△	△	△	▲
4 FEM	▲	▲	▲	△	△	△	△	△	△	△	△	△	△	△	▲
4 NN	▲	▲	▲	△	△	△	△	△	△	△	△	△	△	△	▲

Tabella 7.11: BTD Punto 3 (0.5, 0.5, -0.005).

DOF	$u_{\alpha 1}$	$u_{\beta 1}$	u_{z1}	$u_{\alpha 2}$	$u_{\beta 2}$	u_{z2}	$u_{\alpha 3}$	$u_{\beta 3}$	u_{z3}	$u_{\alpha 4}$	$u_{\beta 4}$	u_{z4}	$u_{\alpha 5}$	$u_{\beta 5}$	u_{z5}
14 FEM	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲
14 NN	▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲
13 FEM	▲	▲	▲	▲	▲	▲	▲	△	▲	△	▲	▲	▲	▲	▲
13 NN	▲	▲	▲	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	△	▲
12 FEM	▲	▲	▲	▲	▲	▲	▲	△	▲	△	▲	▲	△	▲	▲
12 NN	▲	▲	▲	▲	▲	▲	△	△	▲	▲	△	▲	▲	▲	▲
11 FEM	▲	▲	▲	▲	▲	▲	△	△	▲	△	▲	▲	△	▲	▲
11 NN	▲	▲	▲	▲	▲	▲	△	▲	▲	△	△	▲	▲	△	▲
10 FEM	▲	▲	▲	▲	▲	▲	△	△	▲	△	▲	▲	△	△	▲
10 NN	▲	▲	▲	▲	▲	▲	△	△	▲	△	△	▲	▲	△	▲
9 FEM	▲	▲	▲	▲	▲	▲	△	△	▲	△	△	▲	△	△	▲
9 NN	▲	▲	▲	▲	▲	▲	△	△	▲	△	△	▲	△	△	▲
8 FEM	▲	▲	▲	▲	▲	△	△	△	▲	△	△	▲	△	△	▲
8 NN	▲	▲	▲	▲	▲	△	△	△	▲	△	△	▲	△	△	▲
7 FEM	▲	▲	▲	▲	▲	△	△	△	▲	△	△	△	△	△	▲
7 NN	▲	▲	▲	▲	▲	△	△	△	▲	△	△	△	△	△	▲
6 FEM	▲	▲	▲	△	▲	△	△	△	▲	△	△	△	△	△	▲
6 NN	▲	▲	▲	△	▲	△	△	△	▲	△	△	△	△	△	▲
5 FEM	▲	▲	▲	△	△	△	△	△	▲	△	△	△	△	△	▲
5 NN	▲	▲	▲	△	△	△	△	△	▲	△	△	△	△	△	▲
4 FEM	▲	▲	▲	△	△	△	△	△	△	△	△	△	△	△	▲
4 NN	▲	▲	▲	△	△	△	△	△	△	△	△	△	△	△	▲

Si osserva che, a differenza dei valori predetti sul parametro di controllo, la rete neurale riesce a predire accuratamente le distribuzioni di termini attivi, presentando solo poche differenze, evidanziate dal colore rosso, con la controparte valutata dal codice di calcolo strutturale. Questo determina comunque la potenzialità che la rete riesce ad assumere. Probabilmente per un corretto funzionamento su entrambi gli aspetti della prediction, si renderebbe necessario cambiare l'architettura interna per svolgere questo tipo di analisi.
Questo procedimento ha portato miglioramenti rispetto al precedente, anche se si perde l'informazione sugli altri failure index. Questo è attualmente risolvibile pensando di utilizzare delle reti neurali in parallelo, in cui ognuna opera su un singolo indice.
Si considera però intanto questa strada percorribile per riuscire a riconoscere le distribuzioni di termini, pertanto si testa la rete aumentando il numero di punti su cui questa analisi viene richiesta.
Vengono dunque eseguite altre analisi, una con 25 punti ed una con 81 punti, scelti per comodità sempre sulla stessa faccia, in maniera tale da considerare un failure index che fosse sempre non nullo. La rappresentazione dei risultati risulta difficoltosa, vista l'ingente quantità di valori da dover riportare. Viene dunque proposta una rappresentazione che mette a confronto, in ciascun punto analizzato, gli errori percentuali

calcolati tramite codice di calcolo fem e rete neurale. Questi sono riportati sono per alcune numerosità significative di termini attivi. Vengono affiancati i grafici ottenuti dall'analisi su 25 punti con quelli ottenuti dall'analisi su 81 punti, in modo tale da poterne notare le somiglianze. La numerazione presente sull'asse x è rappresentativa del numero identificativo del punto, che differisce tra le due analisi.

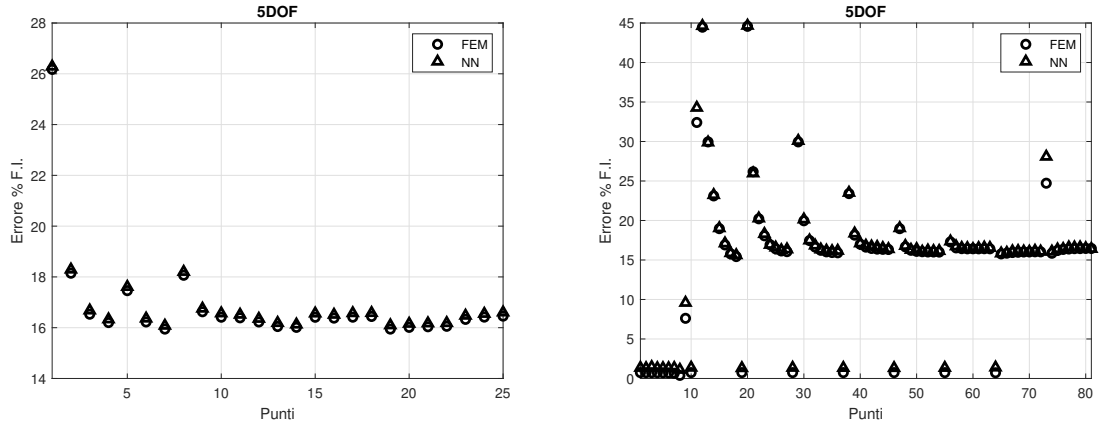


Figura 7.1: Confronto FEM - NN, caso a 5DOF 25 punti ed 81 punti.

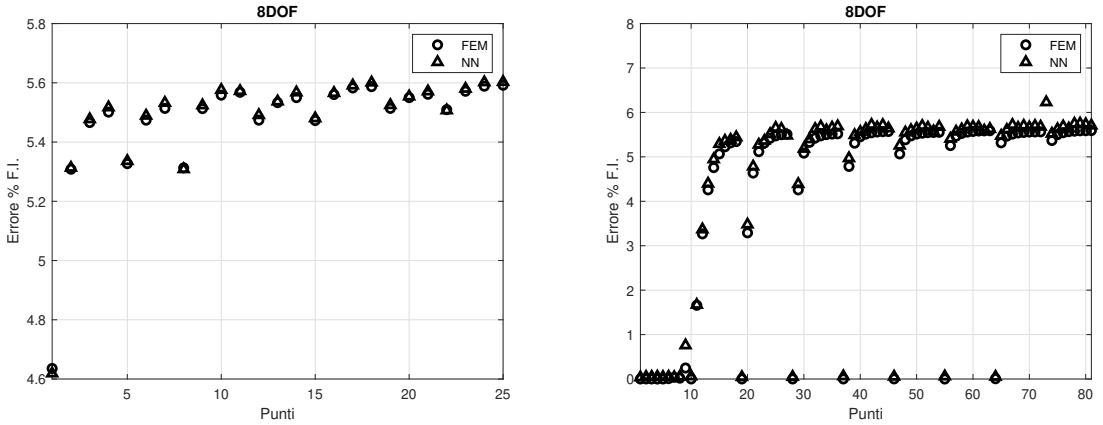


Figura 7.2: Confronto FEM - NN, caso a 8DOF 25 punti ed 81 punti.

Si riesce a notare che gli andamenti ottenuti sono gli stessi che si potevano ricavare anche dall'analisi effettuata su solamente 3 punti della struttura. Gli errori nei casi di basse numerosità sembrano quasi coincidere, mentre in quelle più alte la differenza tra fem e rete neurale è più marcata. L'aspetto importante però è il fatto che la rete neurale riesca a mantenere lo stesso tipo di accuratezza anche aumentando il numero di punti su cui viene richiesto il funzionamento.
Inoltre nelle tabelle 7.12, 7.13 e 7.14, si riporta la distribuzione dei termini attivi nei casi a 5, 8 e 12 DOF, considerando due punti della struttura comuni alle tre analisi effettuate (considerando 3, 25 ed 81 punti) ed effettuando un confronto tra di loro e

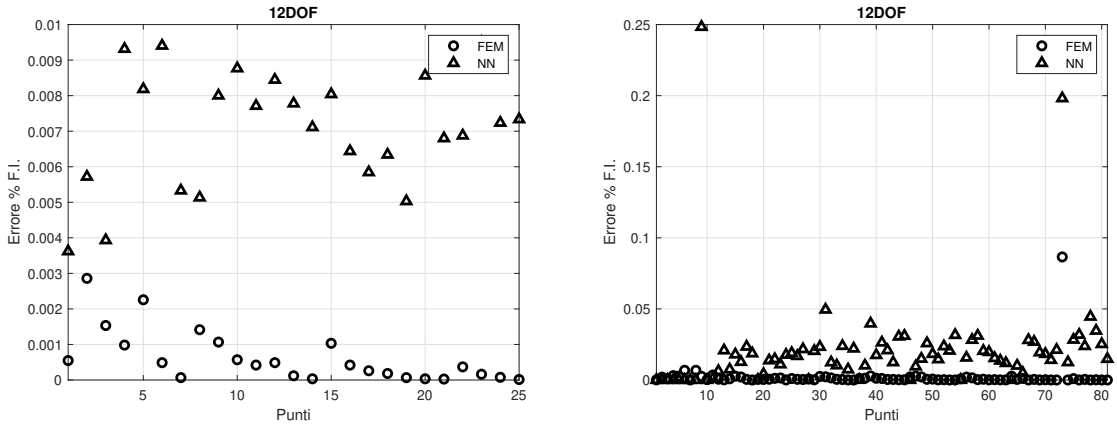


Figura 7.3: Confronto FEM - NN, caso a 12DOF 25 punti ed 81 punti.

tra il riferimento ottenuto dall'analisi tramite il software di calcolo strutturale. Come si può osservare anche la prediction delle distribuzioni di termini effettuate dalla rete neurale non subisce modifiche rilevanti all'aumentare del numero di punti richiesti e continua a mantenere l'ottima accuratezza.

Tabella 7.12: BTD 5 DOF - Confronto analisi FEM - rete neurale a 3, 25, 81 punti.

(0.5, 0.5, -0.005)					3pt.					25pt.					81pt.				
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△
▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲
(0.25, 0.25, -0.005)					3pt.					25pt.					81pt.				
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△
▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△	▲	△	△	△	△
▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲	▲	△	▲	△	▲

Tabella 7.13: BTD 8 DOF - Confronto analisi FEM - rete neurale a 3, 25, 81 punti.

(0.5, 0.5, -0.005)					3pt.					25pt.					81pt.				
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲
(0.25, 0.25, -0.005)					3pt.					25pt.					81pt.				
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△	▲	▲	△	△	△
▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲	▲	△	▲	▲	▲

Tabella 7.14: BTD 12 DOF - Confronto analisi FEM - rete neurale a 3, 25, 81 punti.

(0.5, 0.5, -0.005)					3pt.					25pt.					81pt.				
▲	▲	▲	△	△	▲	▲	△	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲
▲	▲	△	▲	▲	▲	▲	△	△	▲	▲	▲	△	△	▲	▲	▲	△	△	△
▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲
(0.25, 0.25, -0.005)					3pt.					25pt.					81pt.				
▲	▲	▲	△	▲	▲	▲	△	▲	▲	▲	▲	▲	▲	△	▲	▲	▲	▲	▲
▲	▲	△	▲	△	▲	▲	△	△	▲	▲	▲	△	△	▲	▲	▲	△	△	△
▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲	▲

7.3 Variazione dell'output

Vista la difficoltà nella riproduzione dei failure index dovuta al crearsi di valori *nan* ed *inf*, si vuole cercare di riprodurre le tensioni σ, essendo queste alla base del calcolo dei failure index.
Per fare ciò, occorre effettuare alcune modifiche alla rete neurale in utilizzo. In particolare, si è reso necessario rimuovere la funzione di attivazione sigmoide dal layer di ouput, dovuto al fatto che in questo caso i valori non sono più compresi tra zero ed uno e sono possibili anche valori negativi.
La rete neurale così impostata però non riesce a riprodurre bene i valori. I problemi si verificano a causa dell'elevata differenza di ordini di grandezza (alcuni sono 10^9 e altri anche 10^{-3}). Questo si verifica sia nel caso dei differenti punti considerati, sia nel caso di diverse σ (ad esempio xx, yy, zz molto diverse da xy). In tabella 7.15 viene riportato un esempio sulla grande variabilità e scarsa accuratezza dei risultati prodotti, che variano addirittura di diversi ordini di grandezza.

Tabella 7.15: Confronto prediction tensioni su due punti.

ID Punto	σ_{xx} [MPa]	σ_{yy} [MPa]	σ_{zz} [MPa]	σ_{xz} [MPa]	σ_{yz} [MPa]	$\sigma_{xy}[MPa]$
1 - FEM	6.006×10^{9}	8.923×10^{9}	2.338×10^{10}	7.177×10^{4}	9.109×10^{4}	4.613×10^{2}
1 - NN	1.424×10^{6}	2.112×10^{6}	1.426×10^{6}	1.293×10^{4}	9.282×10^{4}	4.568×10^{2}
2 - FEM	3.003×10^{9}	4.459×10^{9}	1.169×10^{10}	1.103×10^{7}	1.726×10^{7}	7.315×10^{7}
2 - NN	8.070×10^{5}	2.142×10^{6}	8.062×10^{5}	2.981×10^{4}	9.598×10^{4}	4.445×10^{6}

A seguito dell'esito negativo prodotto da questa analisi, vengono effettuati test con il fine ci comprendere meglio cosa impedisca alla rete neurale artificiale di ottenere risultati accurati come nelle altre tipologie di analisi effettuate in questo lavoro di tesi. Un primo tentativo ha riguardato i dati di input, ponendosi il problema se in questo caso il 20% dei dati totali forniti alla rete neurale non fosse sufficiente per elaborare le relazioni matematiche interne, ma anche aumentando la numerosità di dati da fornire

alla rete, non sono stati riscontrati miglioramenti di alcun genere, nemmeno utilizzando la totalità dei dati disponibili forniti dal codice di calcolo strutturale.
Altri test sono stati effettuati operando un'operazione di scalatura sui valori delle tensioni, prima di fornirle nella fase di training della rete neurale. Si è riscontrato però che una scalatura uniforme non è sufficiente per ottenere risultati accettabili, pertanto si è optato per un ulteriore test normalizzando i valori delle tensioni rispetto a parametri geometrici della struttura, secondo le seguenti relazioni:

$$\begin{aligned} \bar{\sigma}_{xx} &= \sigma_{xx}\frac{h^2}{a^2 p_z}; \quad \bar{\sigma}_{yy} = \sigma_{yy}\frac{h^2}{a^2 p_z}; \quad \bar{\sigma}_{zz} = \sigma_{zz}\frac{1}{p_z} \\ \bar{\sigma}_{xz} &= \sigma_{xz}\frac{h}{a p_z}; \quad \bar{\sigma}_{yz} = \sigma_{yz}\frac{h}{a p_z}; \quad \bar{\sigma}_{xy} = \sigma_{xy}\frac{h}{a p_z} \end{aligned} \tag{7.5}$$

I risultati di questo test sono riportati nella tabella 7.16 e si nota immediatamente come questi risultino essere anche in questo caso non accettabili.

Tabella 7.16: Confronto prediction tensioni normalizzate su due punti.

ID Punto	σ_{xx} [MPa]	σ_{yy} [MPa]	σ_{zz} [MPa]	σ_{xz} [MPa]	σ_{yz} [MPa]	$\sigma_{xy}[MPa]$
1 - FEM	6.006×10^{9}	8.923×10^{9}	2.338×10^{10}	7.177×10^{4}	9.109×10^{4}	4.613×10^{2}
1 - NN	9.096×10^{5}	2.852×10^{6}	9.0956×10^{1}	-6.013	1.770×10^{1}	1.611×10^{1}
2 - FEM	3.003×10^{9}	4.459×10^{9}	1.169×10^{10}	1.103×10^{7}	1.726×10^{7}	7.315×10^{7}
2 - NN	4.487×10^{5}	1.399×10^{6}	4.487×10^{1}	2.267×10^{2}	3.458×10^{3}	2.175×10^{6}

Successivamente è stato eseguito un ultimo test, scalando ogni sigma di ogni punto, rispetto al valore minimo che sé stessa assume tra tutte le combinazioni previste dall'approccio assiomatico-asintotico. In questa maniera, la rete effettua un training su valori che risultano essere tutti nello stesso intorno di ordine di grandezza. Questo metodo ha portato a risultati molto accurati, quasi da rendere indistinguibili quelli prodotti dalla rete neurale con quelli prodotti dal codice di calcolo fem, però non vengono riportati in questo elaborato, in quanto il tipo di scalatura che è stato eseguito, prevede il conoscere in anticipo tutti i valori che le tensioni possono assumere in ciascun punto di una determinata struttura, andando così a nullificare l'utilità di sviluppare una rete neurale artificiale con lo scopo di ridurre il costo computazionale richiesto. Ulteriori test non sono stati eseguiti, in quanto bisognerebbe andare a modificare l'architettura vera e propria della rete neurale, per trovare quella più adatta ad effettuare questa tipologia di analisi.

Capitolo 8

Conclusioni

Nel lavoro svolto per questa tesi, è stata sviluppata una rete neurale artificiale in ambiente Pyhton che potesse essere integrata nell'ambito dell'ingegneria aerospaziale, in particolare nel settore strutturale. La motivazione dietro a questo tipo di studio si riscontra nell'utilizzo sempre più esteso delle tecnologie di machine learning, in abbinata con il crescere del costo computazionale richiesto ad analisi di tipo strutturale sempre più accurate e riguardanti strutture dall'elevata complessità.
La problematica è stata affrontata affiancando alla rete neurale artificiale, un software di calcolo strutturale fornito dal gruppo di ricerca MUL2 del dipartimento di ingegneria aerospaziale del Politecnico di Torino. Il software si basa sulla formulazione CUF per l'analisi ad elementi finiti, rendendo così facilmente accessibile l'implementazione della rete neurale con le varie analisi che sono state eseguite e permettendo l'applicazione innovativa del metodo assiomatico-asintotico per la determinazione dei Best Theory Diagrams e delle Best Theory Distributions, nelle analisi di tipo global/local.
Inizialmente è stata svolta una fase di training per ottenere familiarità sia con il software di calcolo, sia con la programmazione in machine learning. Una volta consolidate queste basi, si è affrontato lo sviluppo dell'architettura di una rete neurale artificiale, il cui campo di utilizzo corrisponde alla riproduzione degli errori prodotti da differenti ordini di teorie strutturali per ognuna delle prime dieci frequenze naturali della struttura in esame. L'innovazione e la peculiarità di questo utilizzo risiede principalmente nella capacità di ottenere un output multidimensionale, infatti la rete neurale viene allenata contemporaneamente sulle varie frequenze. Questo approccio permette di ridurre notevolmente il costo computazionale, in quanto la rete utilizza solo una piccola percentuale dei dati disponibili, per riuscire a riprodurre fedelmente i BTD di ogni frequenza, risaltando il fatto che la miglior teoria associata ad un'analisi globale considerando la media delle dieci frequenze, non corrisponde alle teorie migliori per ciascuna delle singole frequenze. In una casistica, si è aumentato ulteriormente la dimensione dell'output richiesto, effettuando l'analisi sulle prime venti frequenze e dimostrando l'efficacia della rete neurale anche in questa situazione.
Dopo aver confermato la validità della rete neurale in questo ambito, si è deciso di applicarla per uno studio di tipo global/local per l'individuazione delle migliori distribuzioni di teorie strutturali sugli elementi di cui è composta la mesh della struttura. In questo caso la rete neurale è stata modificata per gestire una diversa quantità di dati di input ed è stata testata un'applicazione riducendo della metà i dati totali forniti. I

risultati ottenuti sono riguardanti le distribuzioni di teorie, anche in questo caso per ciascuna delle prime dieci frequenze naturali, e si sono rilevati essere notevolmente interessanti, soprattutto dovuto al fatto che questo tipo di analisi compiute dal software di calcolo necessitavano diverse ore per essere completate, mentre l'utilizzo della rete neurale ha portato a risultati molto accurati in tempistiche notevolmente ridotte, rendendo sufficiente l'impiego di alcune decine di minuti.
L'ultima applicazione affrontata nella tesi, riguarda l'utilizzo della rete neurale per la determinazione dei failure index e delle tensioni su diversi punti della struttura. In questo caso, la rete neurale viene fatta funzionare con l'aggiunta di una dimensione, infatti il multi-input ed il multi-output sono associati ad ogni singolo punto e le fasi di training e di prediction avvengono contemporaneamente sulla totalità dei punti considerati. In questo caso però vi è un peggioramento dei risultati ottenuti. L'applicazione per la riproduzione delle tensioni risulta insoddisfacente senza andare a modificare l'architettura della rete, mentre nella riproduzione dei failure index si riesce ad ottenere risultati buoni sulla riproduzione delle distribuzioni di termini attivi nei BTD, a fronte di una minore accuratezza sui valori degli errori e tutto questo considerando un solo failure index per ogni analisi, anziché una moltitudine.

8.1 Prospettive future

Quelle viste in questo lavoro di tesi sono solo alcune delle possibili applicazioni che le reti neurali artificiali possono avere nel campo delle analisi strutturali. La validità dei risultati ottenuti conferma il raggiungimento dell'obiettivo di diminuire costo computazionale e tempo di calcolo, mantenendo un'accuratezza accettabile. Tutto questo è stato possibile con lo sviluppo di una rete neurale efficiente, che in futuro potrà indubbiamente essere modificata nella sua architettura per ottenerne un'applicazione estesa ad ulteriori tipologie di analisi. Ad ogni argomento, la sua applicazione è stata effettuata considerando diverse configurazioni sia geometriche che di vincolo della struttura in esame, dimostrando la capacità di produrre risultati accurati a prescindere da queste caratteristiche. Le reti neurali artificiali presentano dunque una grande potenzialità, che andrà investigata ulteriormente per comprendere fino a che punto si può mantenere l'accuratezza riscontrata, andando ad aumentare oltre le casistiche, anche la quantità di risultati producibili. Una possibile prospettiva futura infatti è che saranno sviluppate delle accurate reti neurali artificiali, allenate con un'ingente mole di dati, in grado di sostituire l'utilizzo di elaborati software di calcolo strutturale in svariate occasioni, permettendo dunque un risparmio di tempo e costi.

Appendice A

Grafici BTD

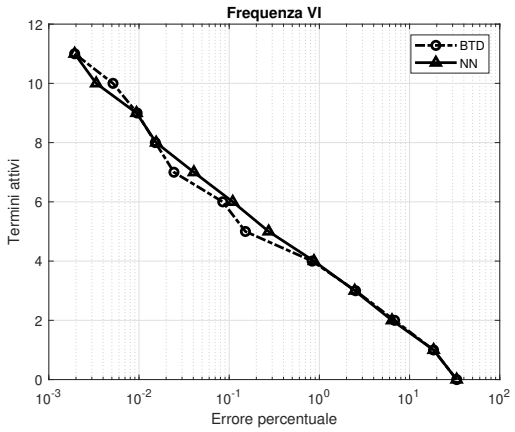


Figura A.1: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenza VI.

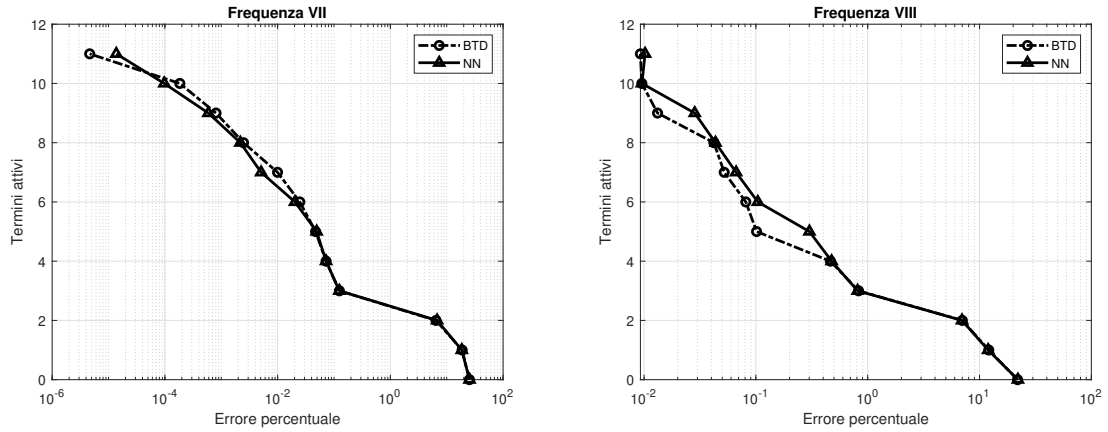


Figura A.2: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze VII e VIII.

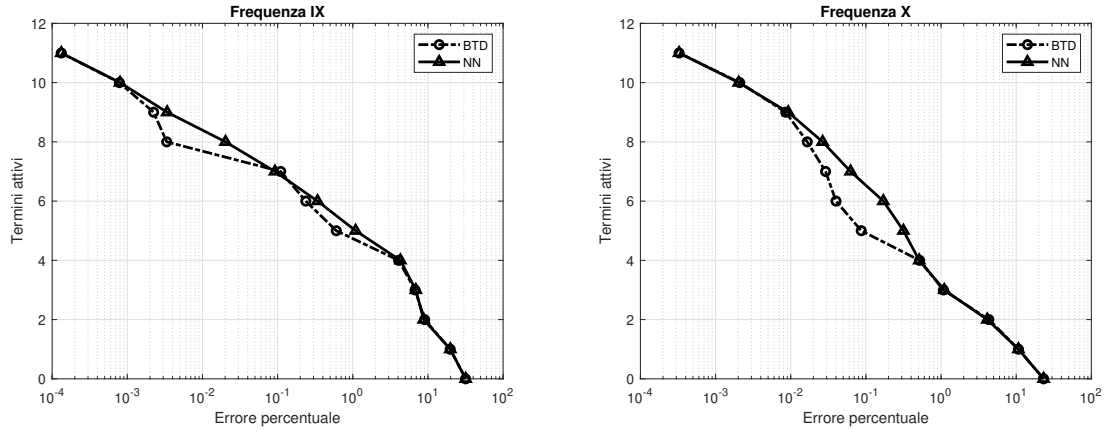


Figura A.3: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze IX e X.

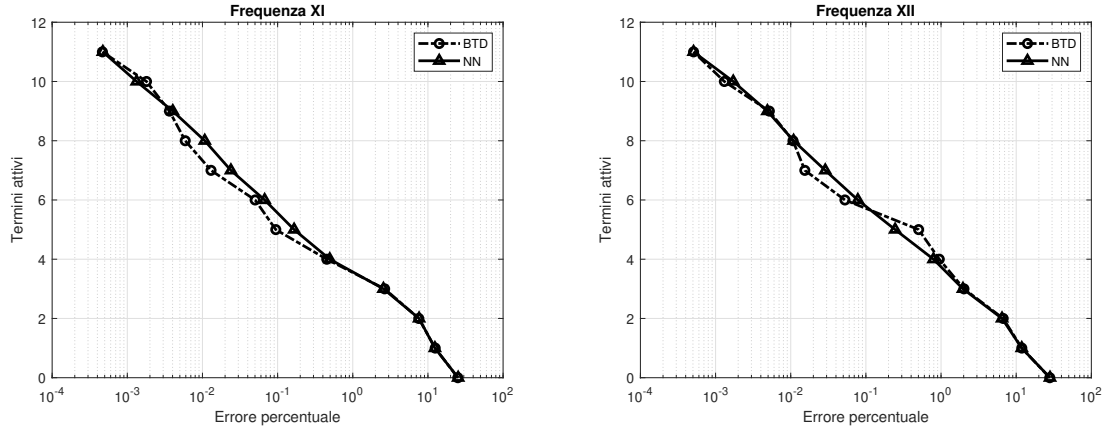


Figura A.4: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze XI e XII.

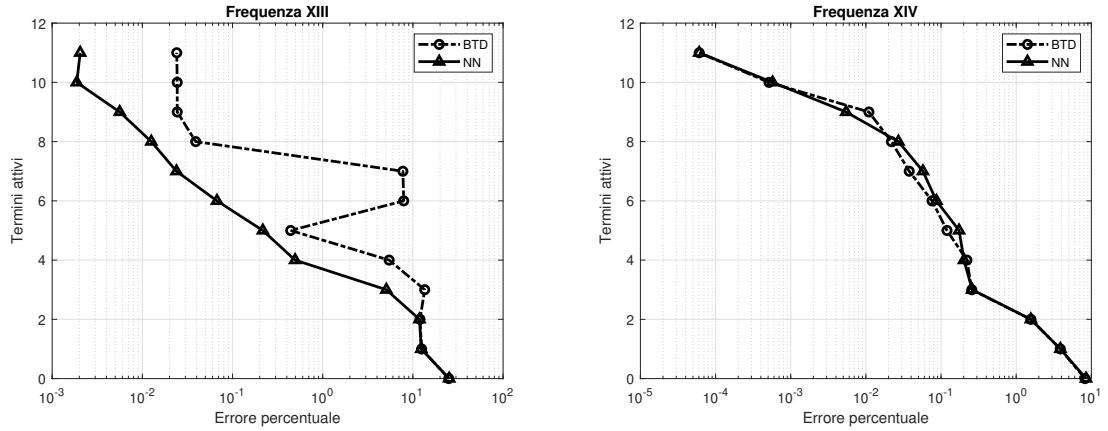


Figura A.5: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze XIII e XIV.

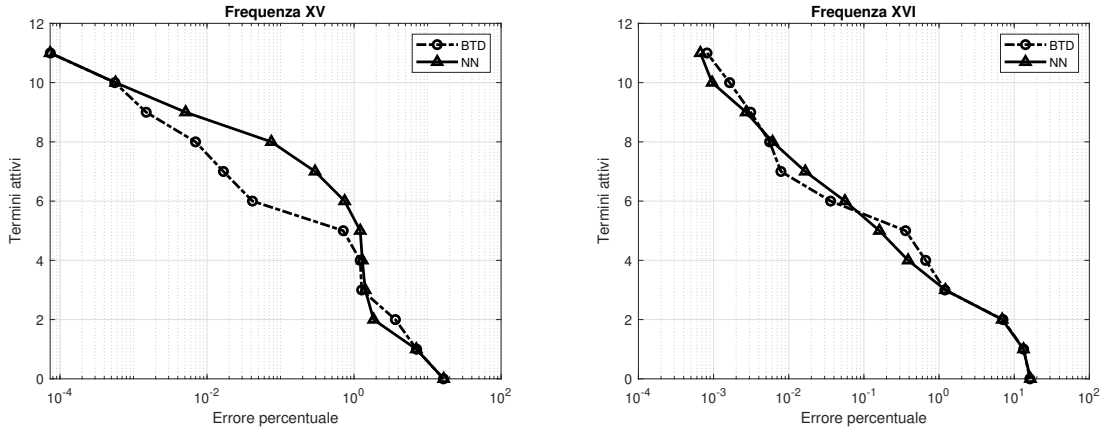


Figura A.6: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze XV e XVI.

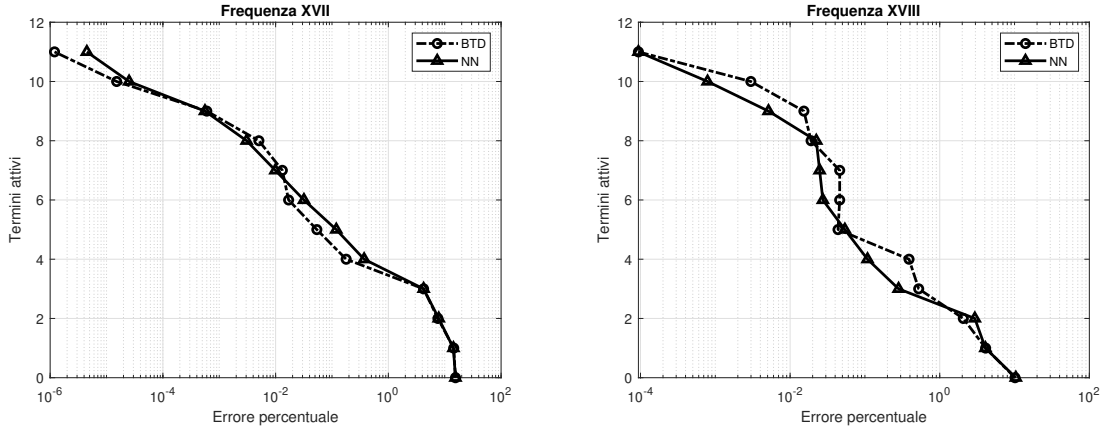


Figura A.7: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze XVII e XVIII.

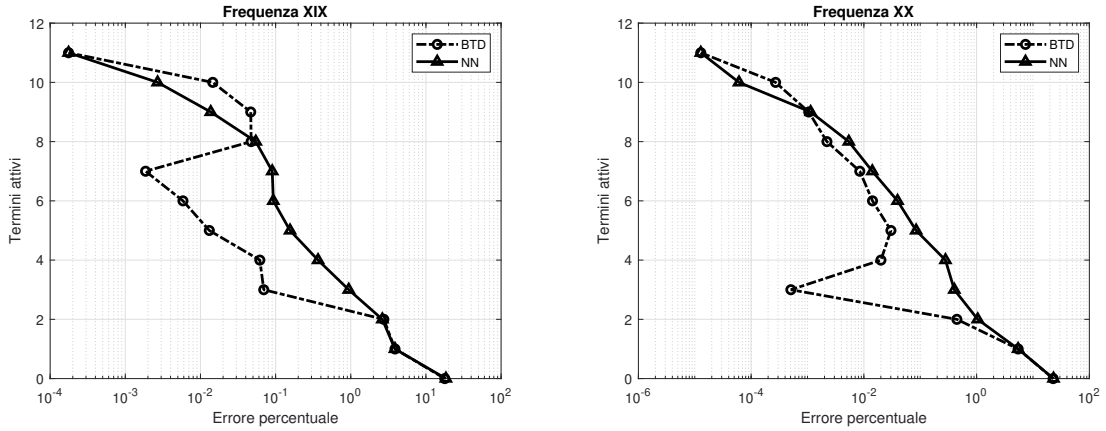


Figura A.8: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=2 - Frequenze XIX e XX.

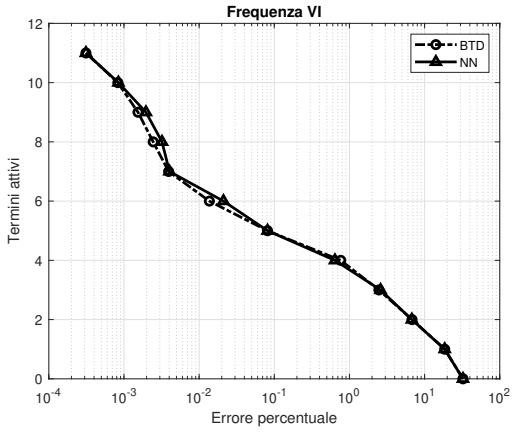


Figura A.9: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5 - Frequenza VI.

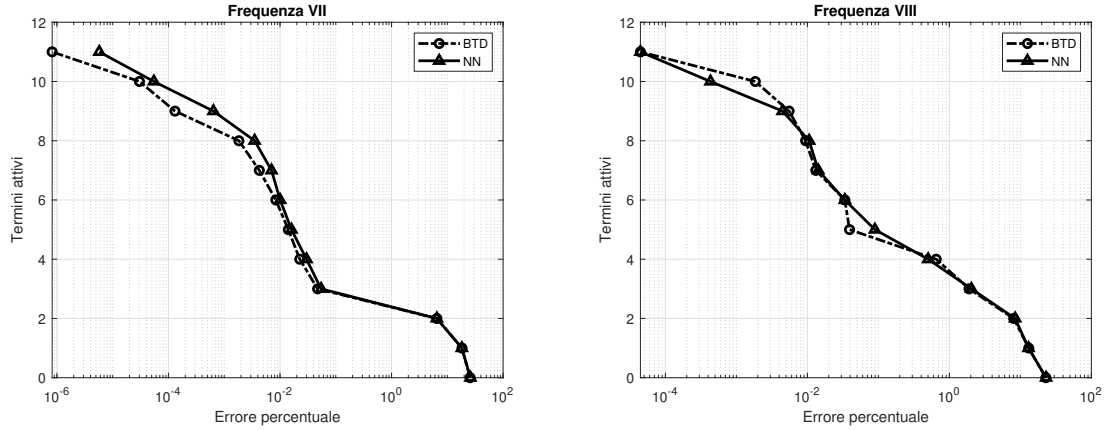


Figura A.10: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5 - Frequenze VII e VIII.

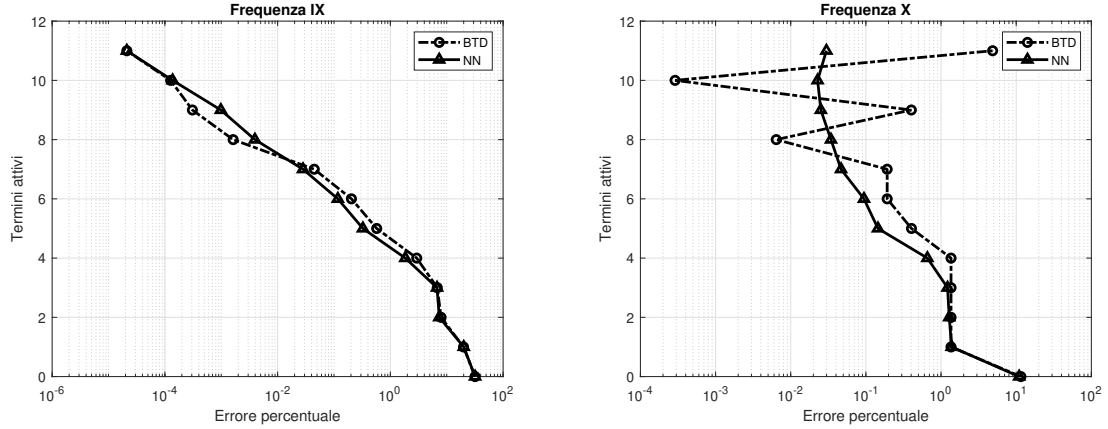


Figura A.11: Piastra semplicemente appoggiata, 0/90/0, a/h=10, R/a=5 - Frequenze IX e X.

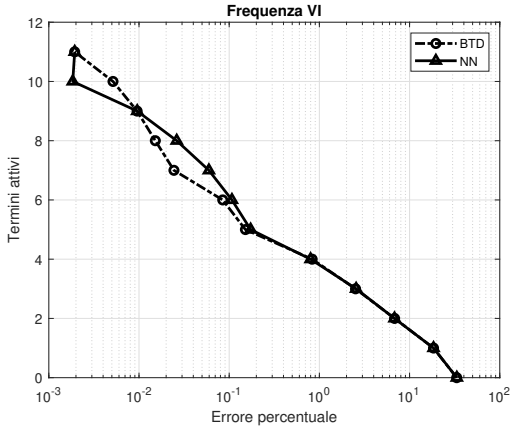


Figura A.12: Piastra semplicemente appoggiata, 0/90/0, a/h=10 - Frequenza VI.

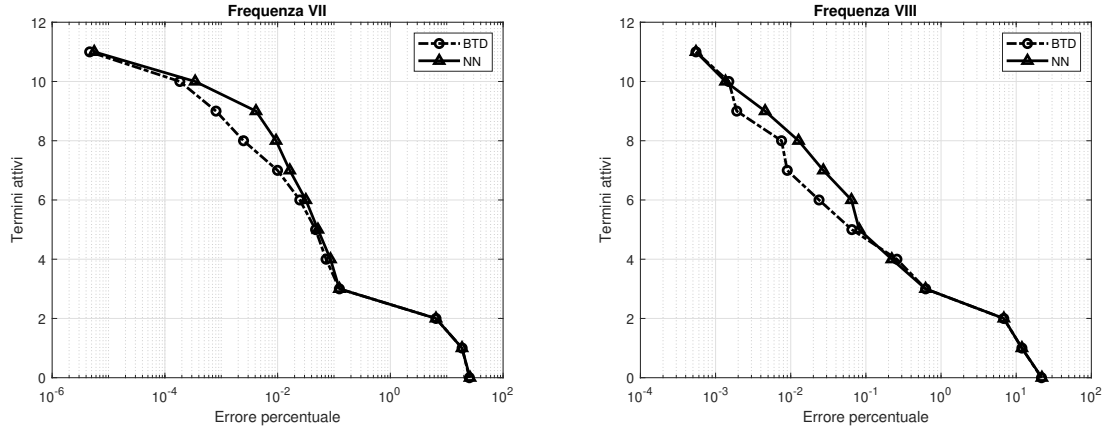


Figura A.13: Piastra semplicemente appoggiata, 0/90/0, a/h=10 - Frequenze VII e VIII.

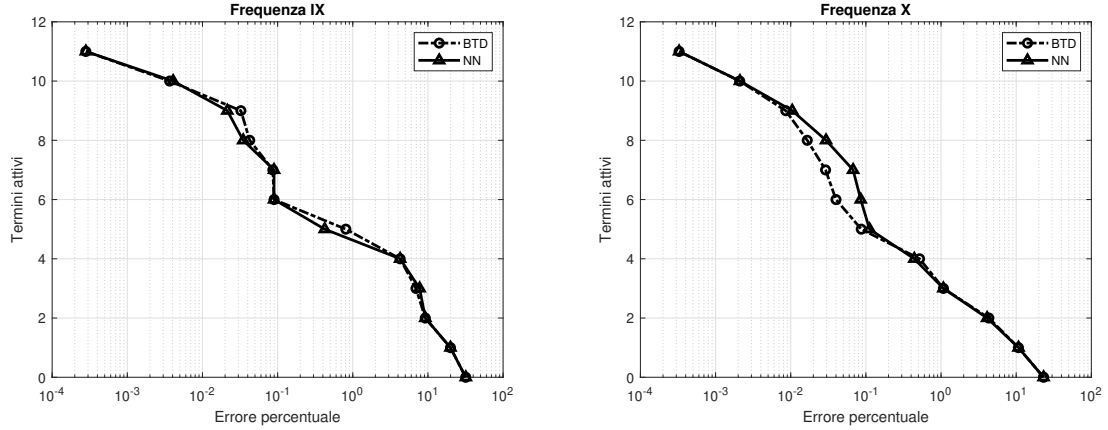


Figura A.14: Piastra semplicemente appoggiata, 0/90/0, a/h=10 - Frequenze IX e X.

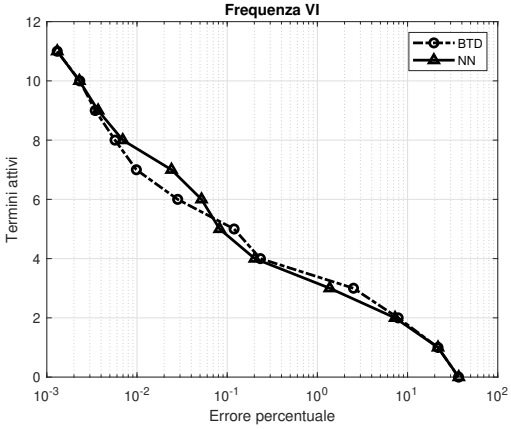


Figura A.15: Piastra clamped - free, 0/90/0, a/h=4, R/a=10 - Frequenza VI.

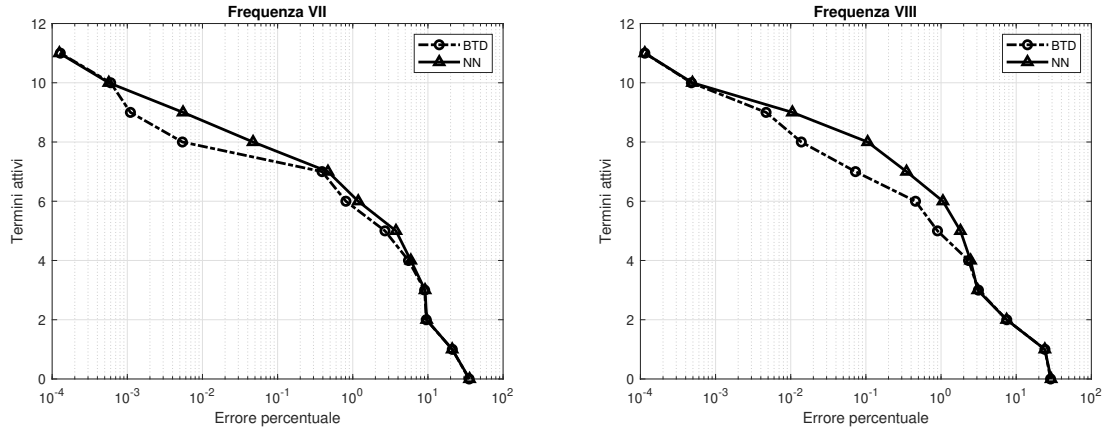


Figura A.16: Piastra clamped - free, 0/90/0, a/h=4, R/a=10 - Frequenze VII e VIII.

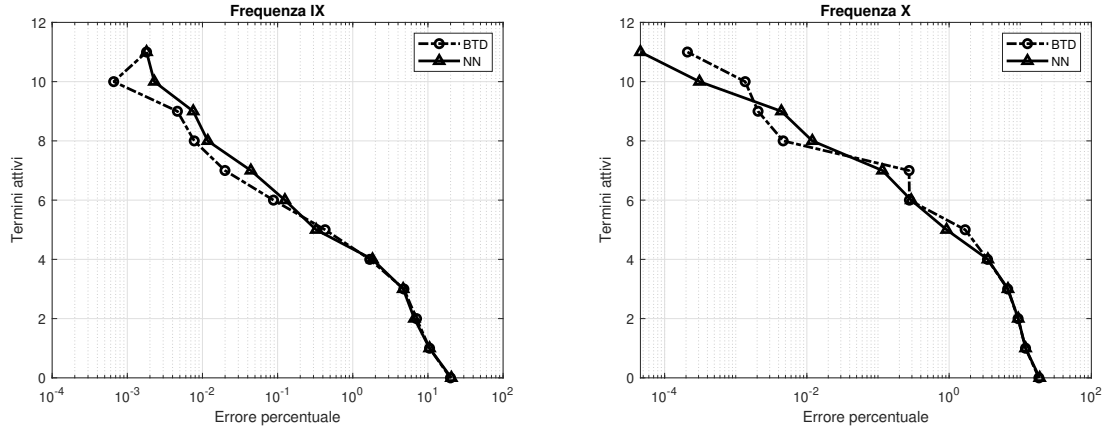


Figura A.17: Piastra clamped - free, 0/90/0, a/h=4, R/a=10 - Frequenze IX e X.

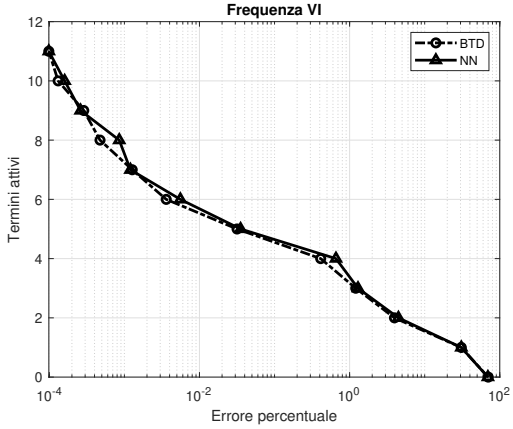


Figura A.18: Piastra clamped - free, 0/90/0, a/h=20, R/a=10 - Frequenza VI.

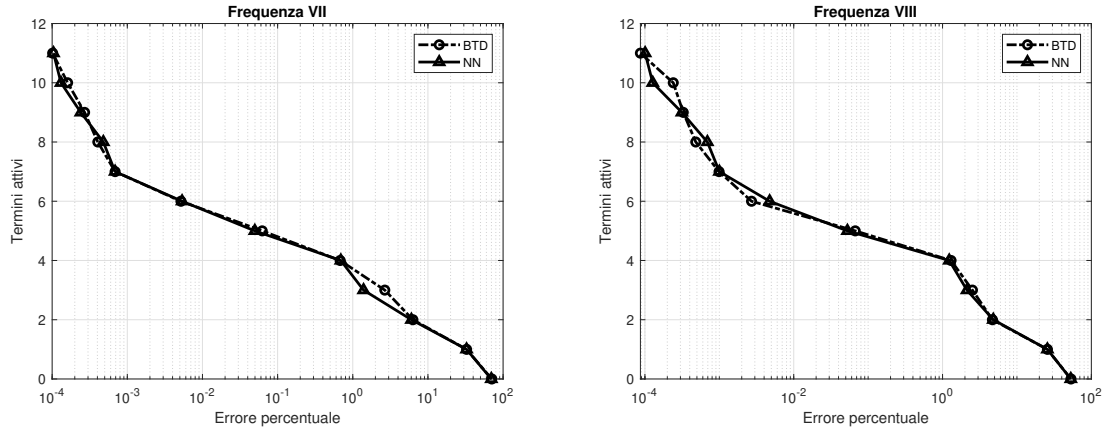


Figura A.19: Piastra clamped - free, 0/90/0, a/h=20, R/a=10 - Frequenze VII e VIII.

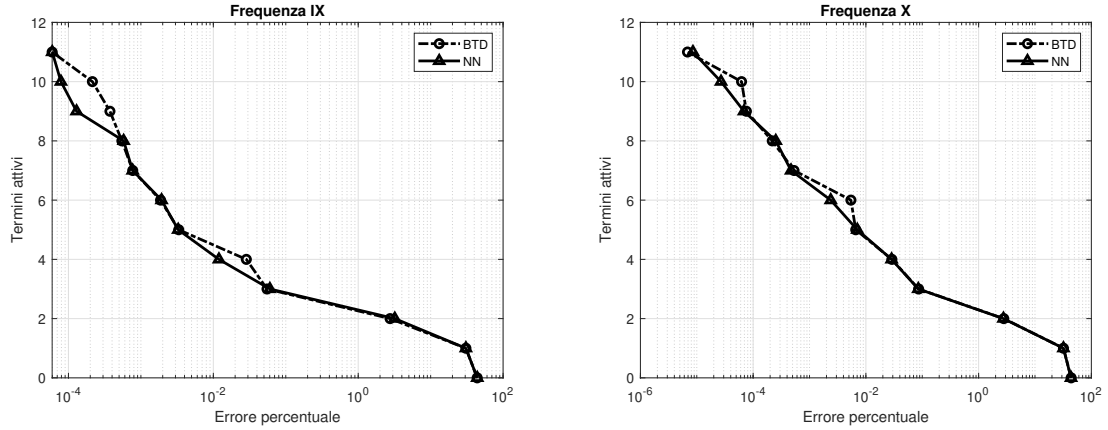
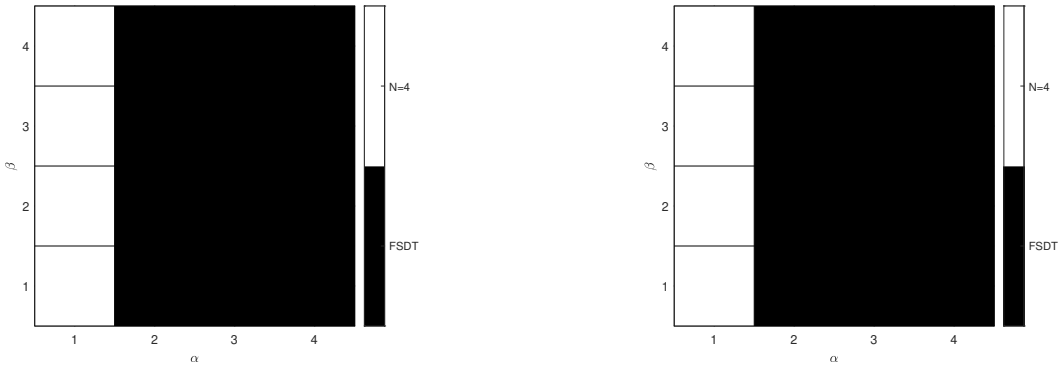


Figura A.20: Piastra clamped - free, 0/90/0, a/h=20, R/a=10 - Frequenze IX e X.

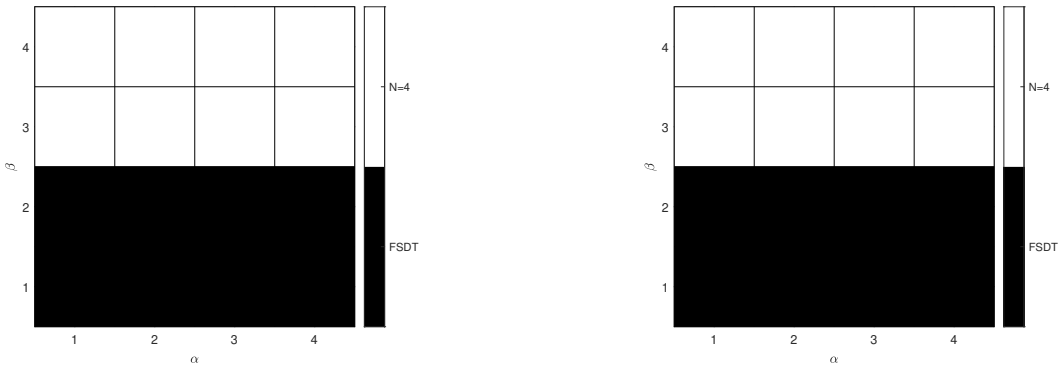
Appendice B

Grafici Global/Local



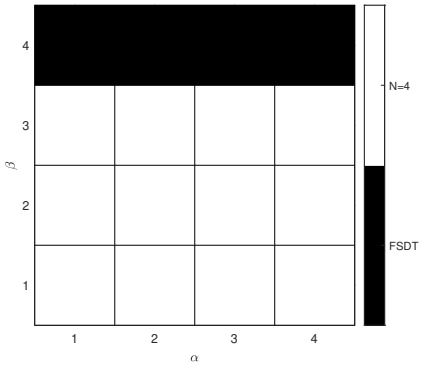
(a) FEM, Err=10.994%. (b) NN, Err=10.953%.

Figura B.1: Frequenza VI - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.

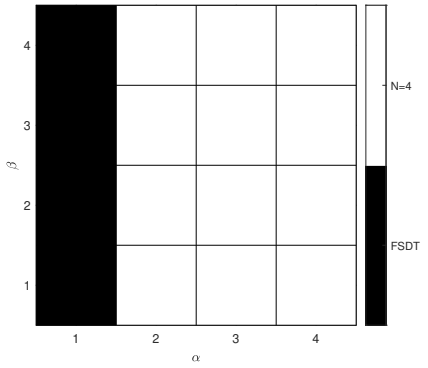


(a) FEM, Err=7.531%. (b) NN, Err=7.454%.

Figura B.2: Frequenza VI - Otto N4, S-S, 0/90/0, a/h=10, R/a=5.

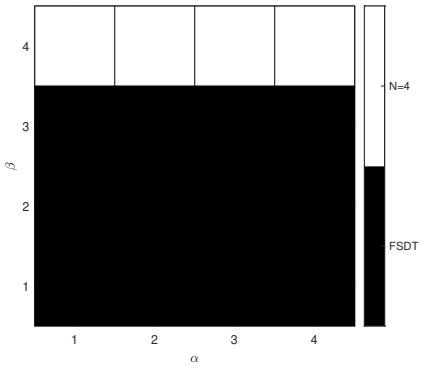


(a) FEM, Err=4.059%.

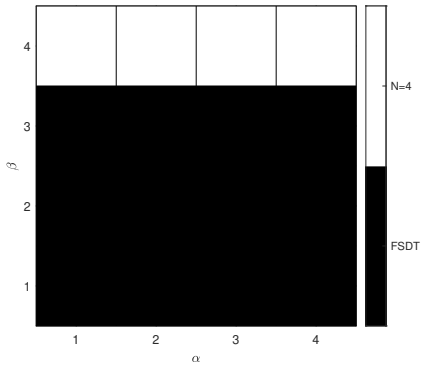


(b) NN, Err=3.877%.

Figura B.3: Frequenza VI - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.

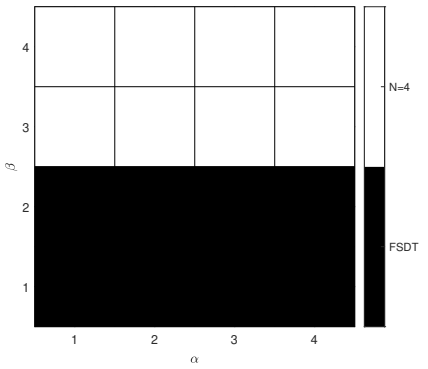


(a) FEM, Err=7.689%.

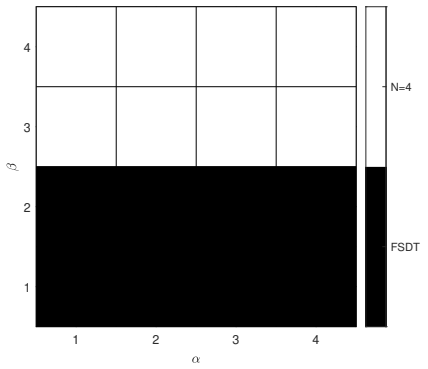


(b) NN, Err=7.735%.

Figura B.4: Frequenza VII - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.

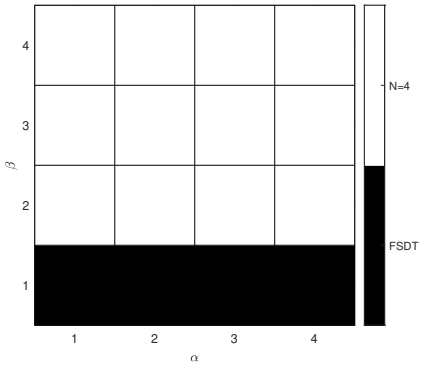


(a) FEM, Err=2.200%.

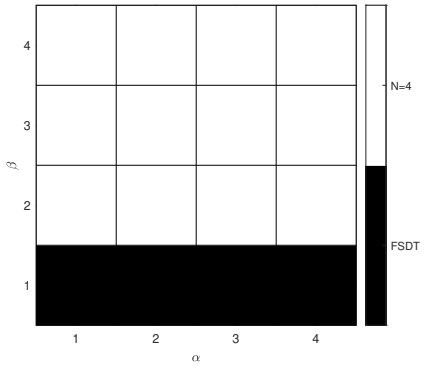


(b) NN, Err=2.209%.

Figura B.5: Frequenza VII - Otto N4, S-S, 0/90/0, a/h=10, R/a=5.

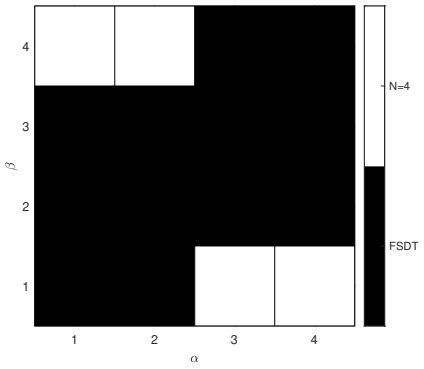


(a) FEM, Err=0.480%.

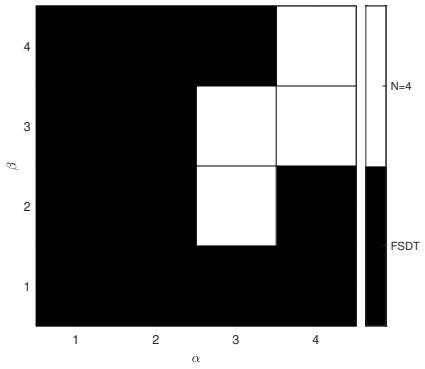


(b) NN, Err=0.498%.

Figura B.6: Frequenza VII - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.

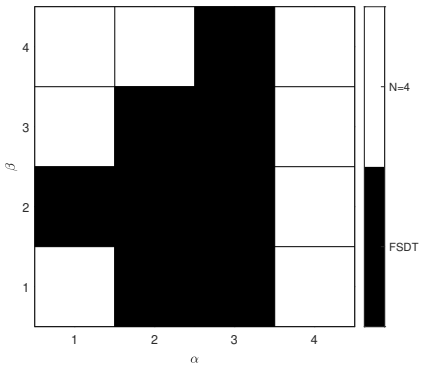


(a) FEM, Err=4.482%.

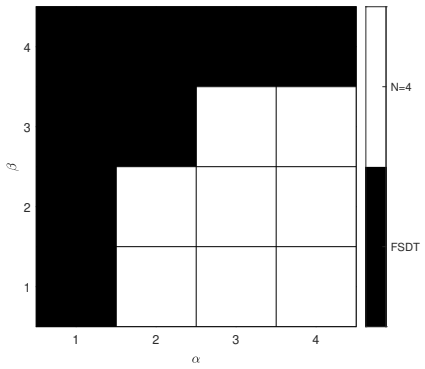


(b) NN, Err=4.946%.

Figura B.7: Frequenza VIII - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.

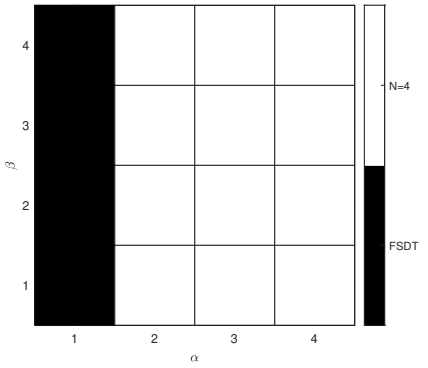


(a) FEM, Err=0.501%.

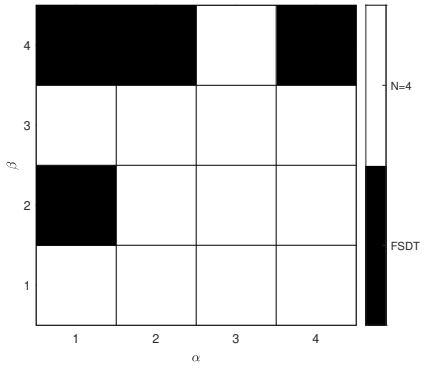


(b) NN, Err=1.731%.

Figura B.8: Frequenza VIII - Otto N4, S-S, 0/90/0, a/h=10, R/a=5.

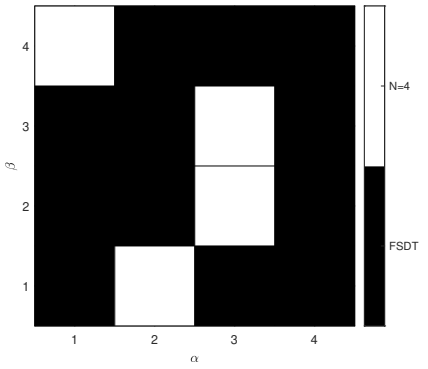


(a) FEM, Err=0.402%.

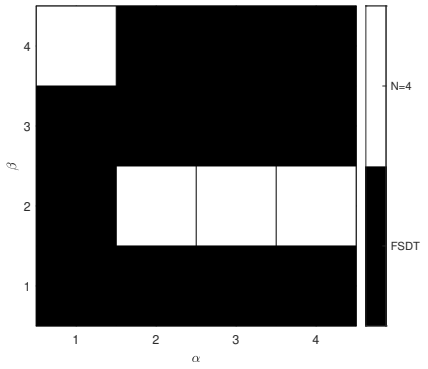


(b) NN, Err=0.295%.

Figura B.9: Frequenza VIII - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.

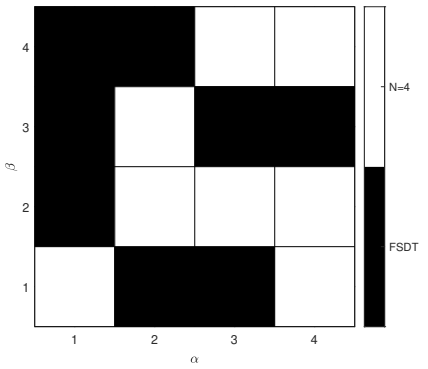


(a) FEM, Err=7.035%.

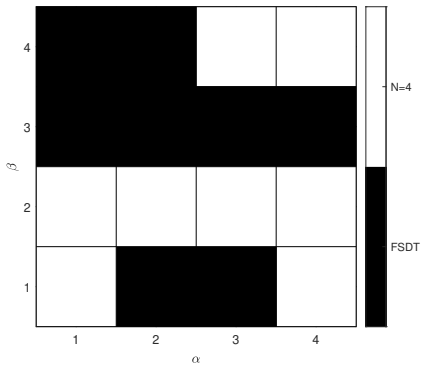


(b) NN, Err=11.516%.

Figura B.10: Frequenza IX - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.

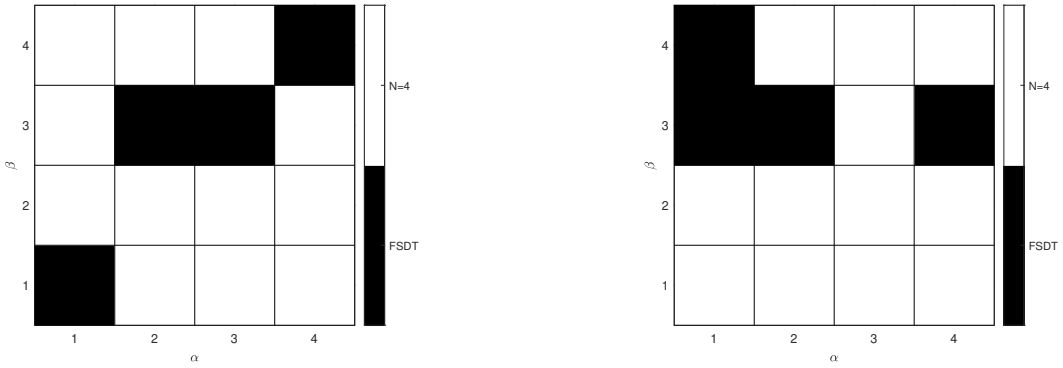


(a) FEM, Err=1.211%.



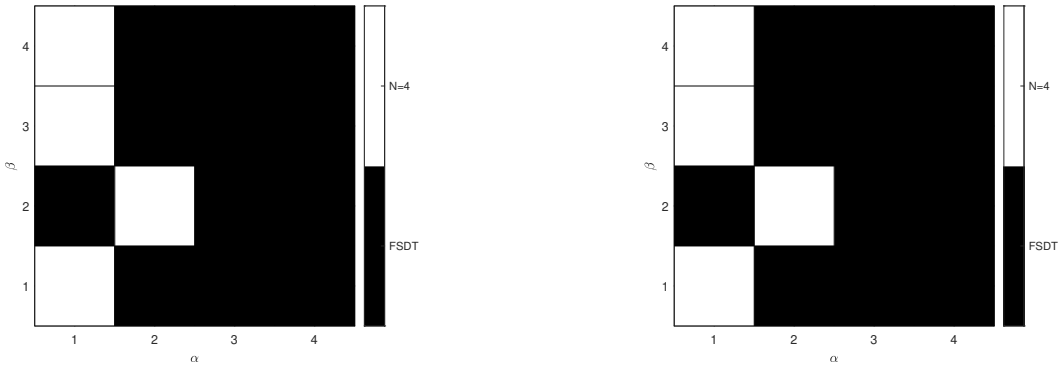
(b) NN, Err=3.018%.

Figura B.11: Frequenza IX - Otto N4, S-S, 0/90/0, a/h=10, R/a=5.



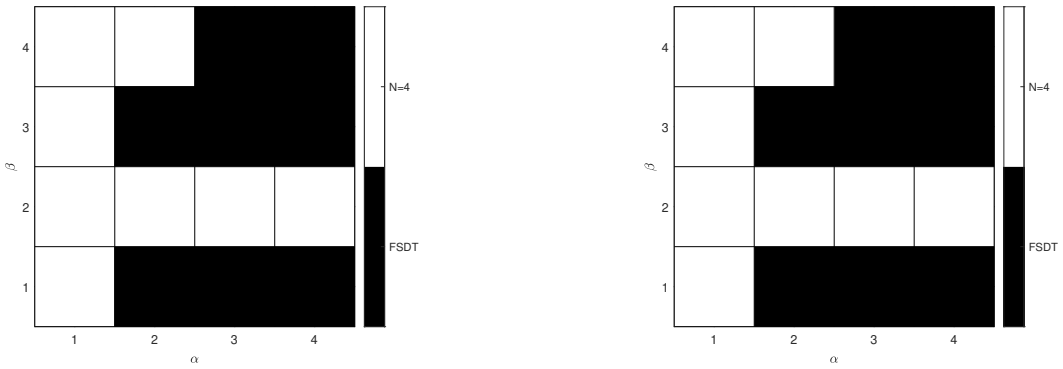
(a) FEM, Err=0.873%. (b) NN, Err=0.839%.

Figura B.12: Frequenza IX - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.



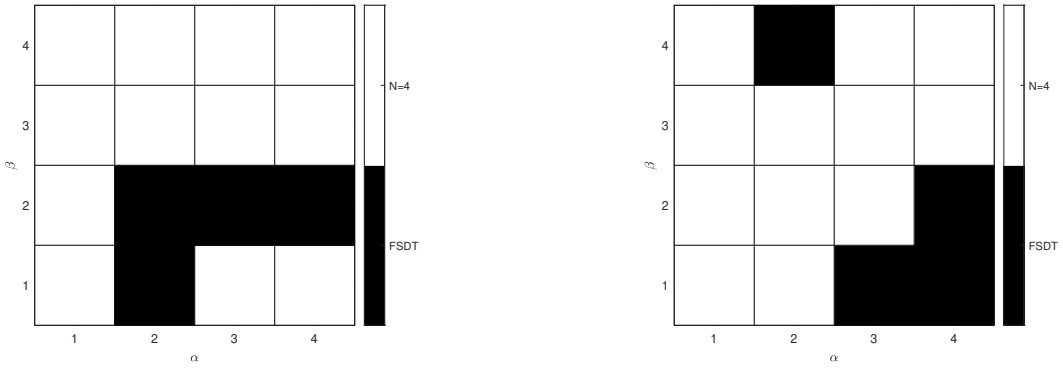
(a) FEM, Err=4.140%. (b) NN, Err=4.138%.

Figura B.13: Frequenza X - Quattro N4, S-S, 0/90/0, a/h=10, R/a=5.



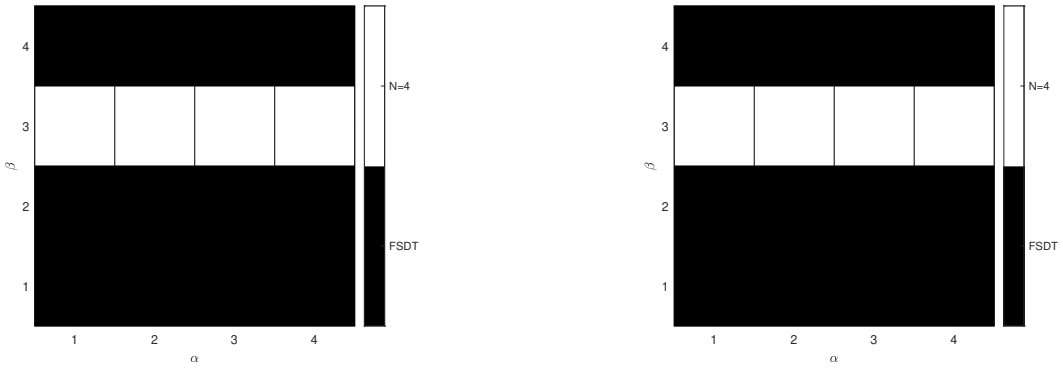
(a) FEM, Err=5.327%. (b) NN, Err=5.301%.

Figura B.14: Frequenza X - Otto N4, S-S, 0/90/0, a/h=10, R/a=5.



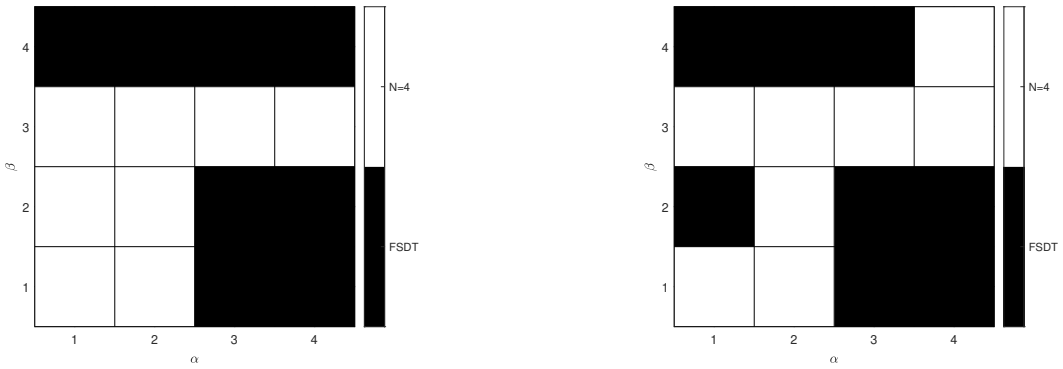
(a) FEM, Err=4.156%. (b) NN, Err=4.370%.

Figura B.15: Frequenza X - Dodici N4, S-S, 0/90/0, a/h=10, R/a=5.



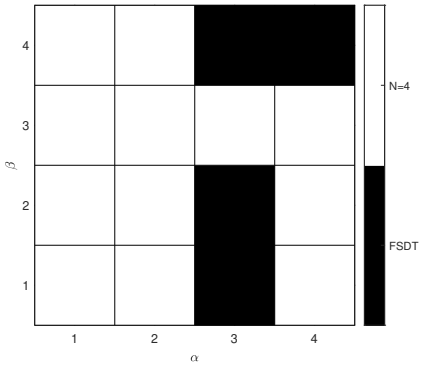
(a) FEM, Err=2.512%. (b) NN, Err=2.504%.

Figura B.16: Frequenza VI - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5.

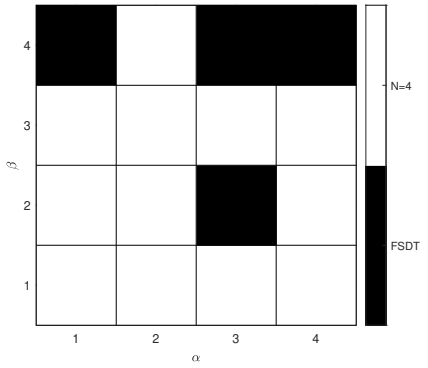


(a) FEM, Err=1.648%. (b) NN, Err=1.657%.

Figura B.17: Frequenza VI - Otto N4, S-S, 0/90/0, a/h=100, R/a=5.

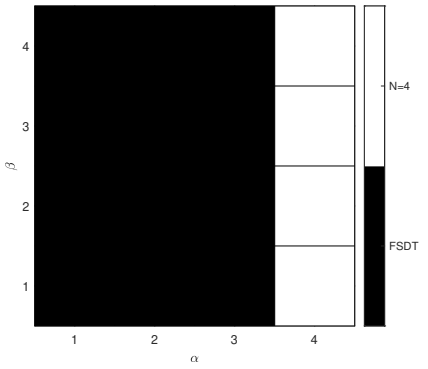


(a) FEM, Err=0.804%.

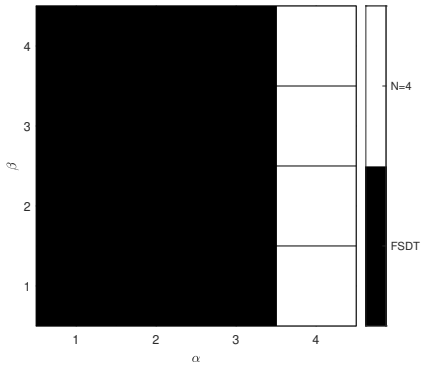


(b) NN, Err=0.785%.

Figura B.18: Frequenza VI - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5.

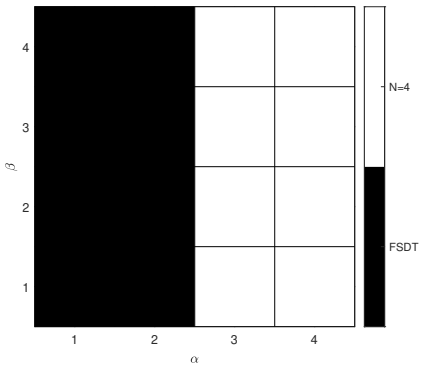


(a) FEM, Err=1.946%.

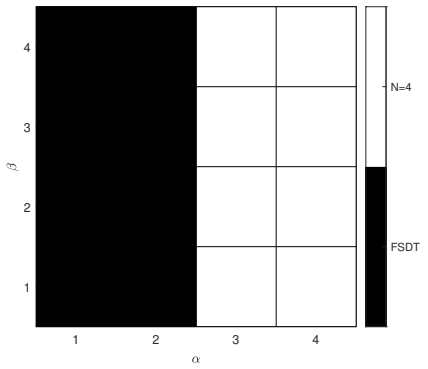


(b) NN, Err=1.953%.

Figura B.19: Frequenza VII - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5.

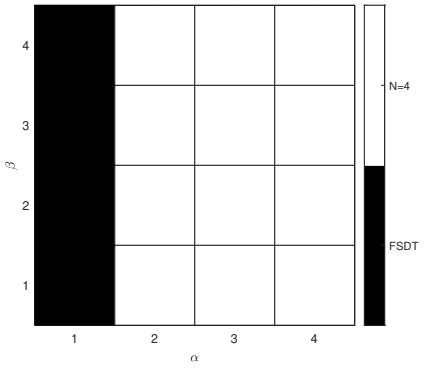


(a) FEM, Err=0.706%.

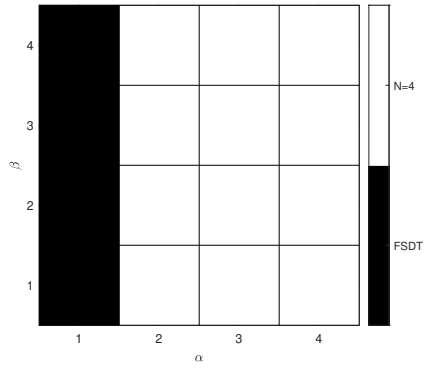


(b) NN, Err=0.729%.

Figura B.20: Frequenza VII - Otto N4, S-S, 0/90/0, a/h=100, R/a=5.

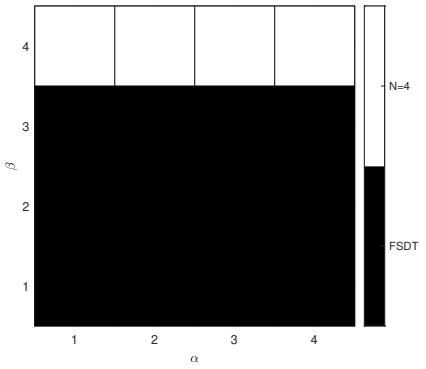


(a) FEM, Err=0.120%.

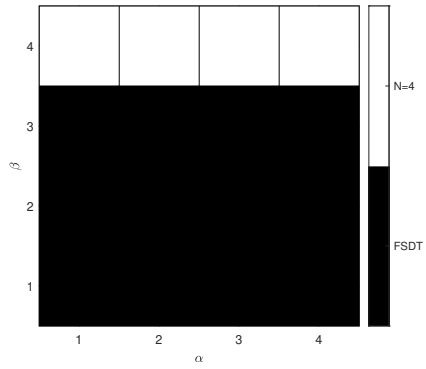


(b) NN, Err=0.123%.

Figura B.21: Frequenza VII - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5.

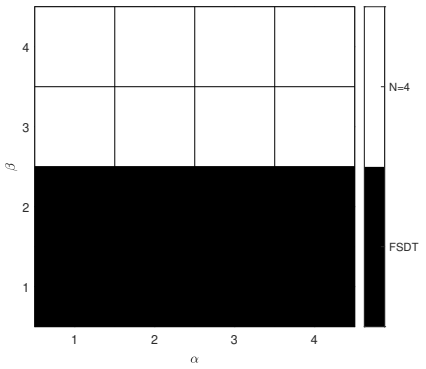


(a) FEM, Err=2.024%.

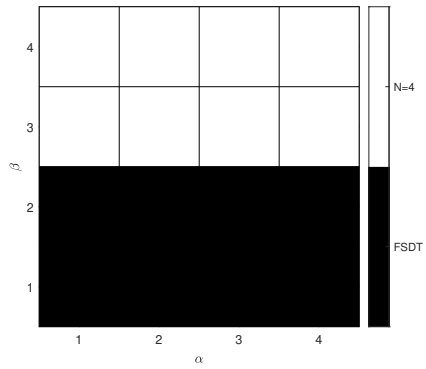


(b) NN, Err=2.029%.

Figura B.22: Frequenza IX - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5.

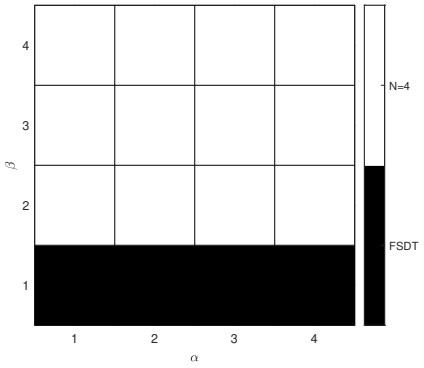


(a) FEM, Err=0.526%.

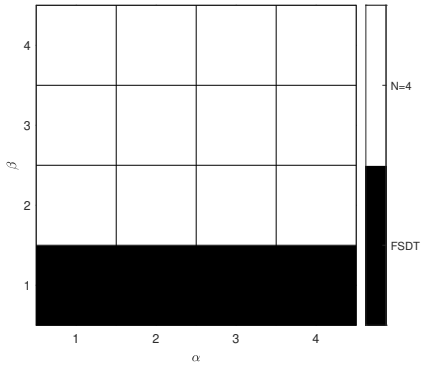


(b) NN, Err=0.527%.

Figura B.23: Frequenza IX - Otto N4, S-S, 0/90/0, a/h=100, R/a=5.

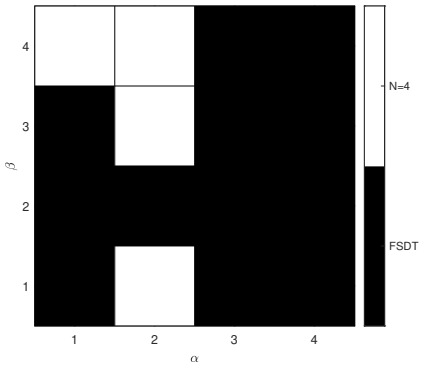


(a) FEM, Err=0.117%.

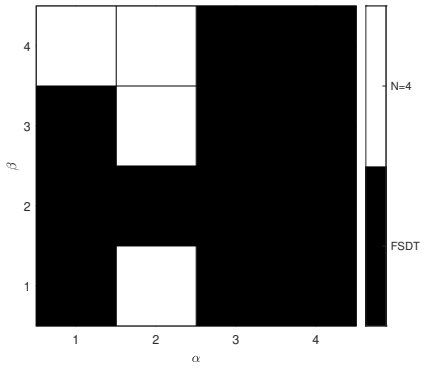


(b) NN, Err=0.113%.

Figura B.24: Frequenza IX - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5.

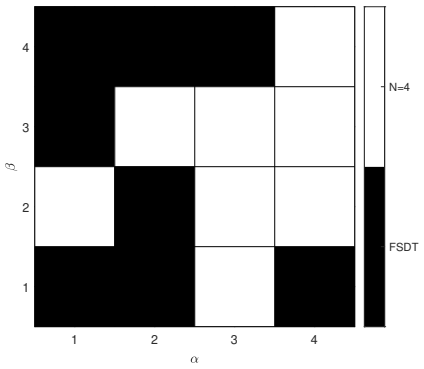


(a) FEM, Err=0.432%.

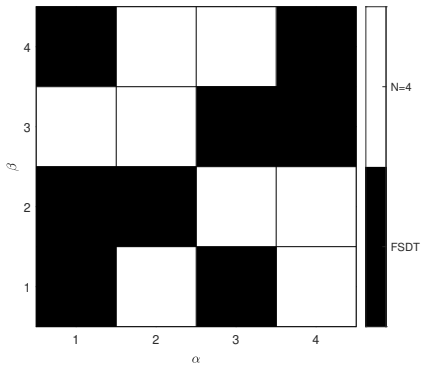


(b) NN, Err=0.432%.

Figura B.25: Frequenza X - Quattro N4, S-S, 0/90/0, a/h=100, R/a=5.

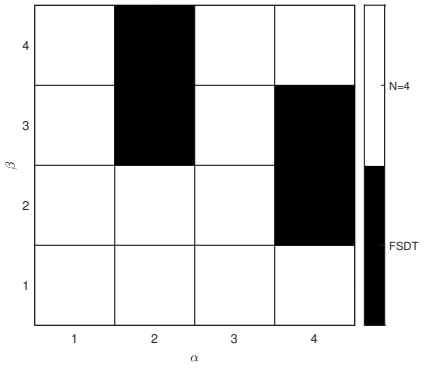


(a) FEM, Err=0.001%.

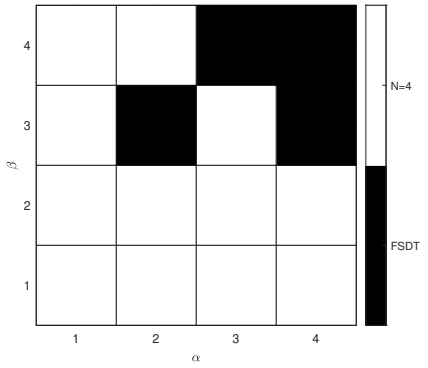


(b) NN, Err=0.021%.

Figura B.26: Frequenza X - Otto N4, S-S, 0/90/0, a/h=100, R/a=5.

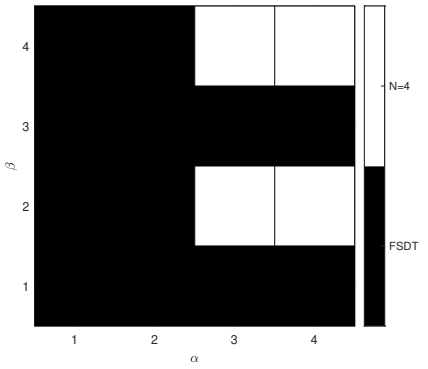


(a) FEM, Err=0.115%.

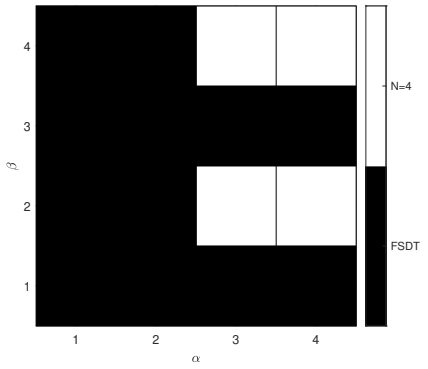


(b) NN, Err=0.085%.

Figura B.27: Frequenza X - Dodici N4, S-S, 0/90/0, a/h=100, R/a=5.

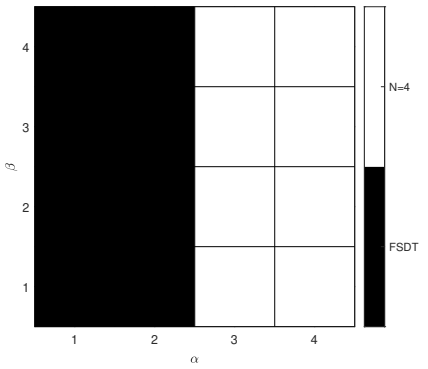


(a) FEM, Err=10.594%.

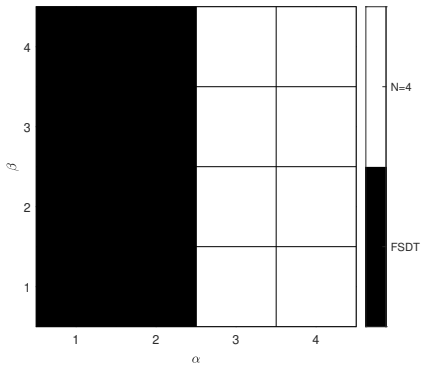


(b) NN, Err=10.660%.

Figura B.28: Frequenza VI - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5.

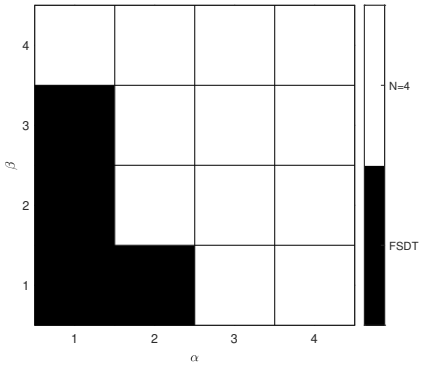


(a) FEM, Err=6.857%.

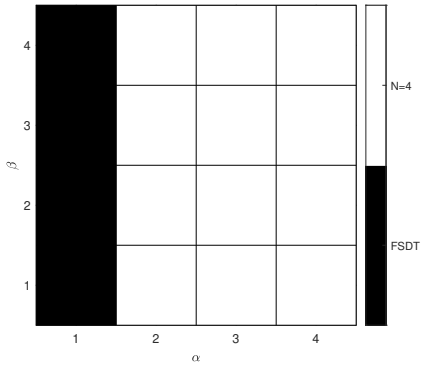


(b) NN, Err=6.741%.

Figura B.29: Frequenza VI - Otto N4, C-F, 0/90/0, a/h=10, R/a=5.

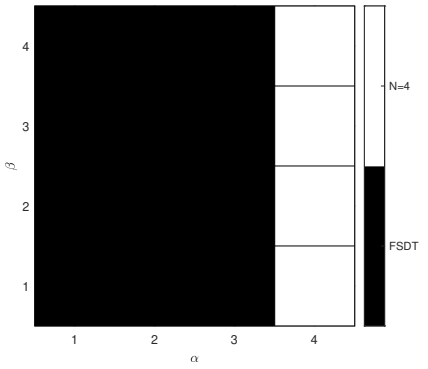


(a) FEM, Err=3.555%.

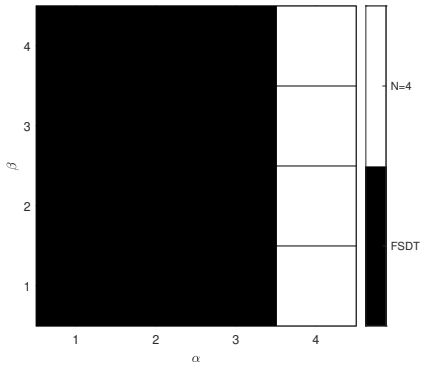


(b) NN, Err=3.396%.

Figura B.30: Frequenza VI - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5.

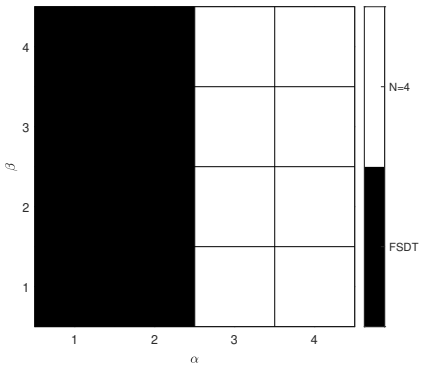


(a) FEM, Err=5.881%.

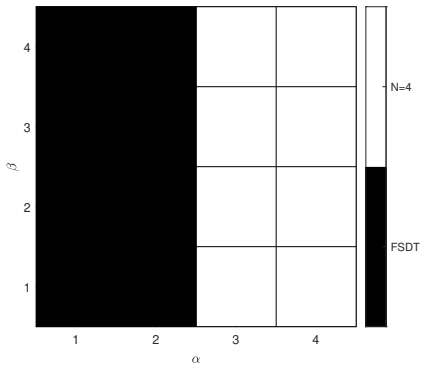


(b) NN, Err=5.932%.

Figura B.31: Frequenza VII - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5.

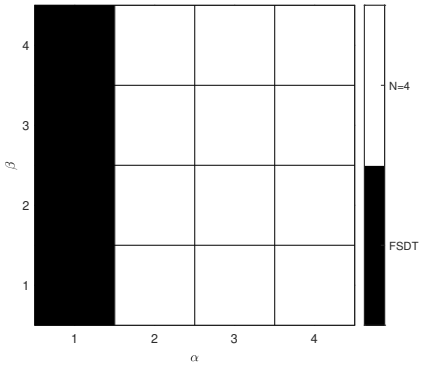


(a) FEM, Err=2.544%.

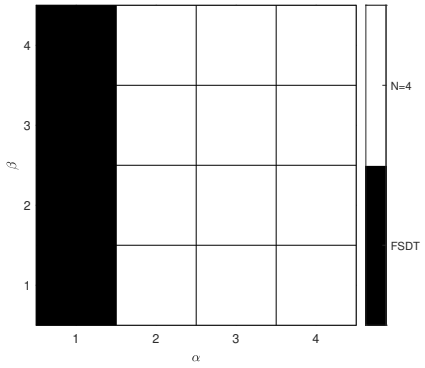


(b) NN, Err=2.568%.

Figura B.32: Frequenza VII - Otto N4, C-F, 0/90/0, a/h=10, R/a=5.

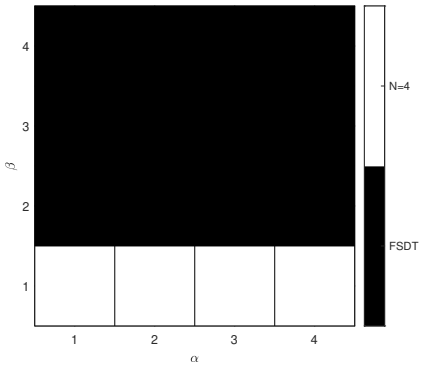


(a) FEM, Err=0.798%.

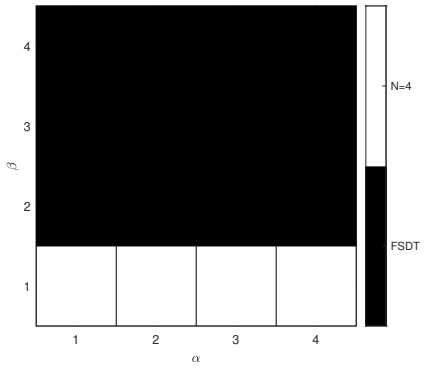


(b) NN, Err=0.647%.

Figura B.33: Frequenza VII - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5.

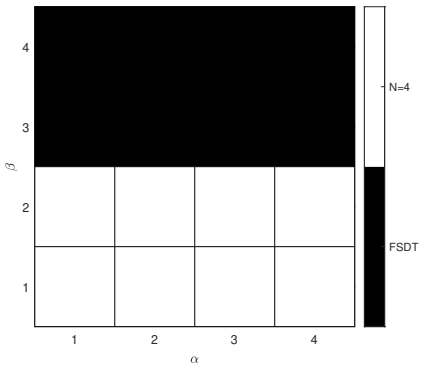


(a) FEM, Err=5.595%.

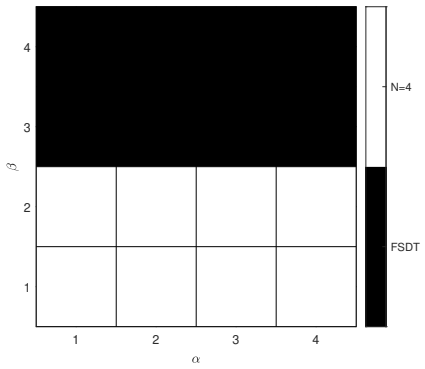


(b) NN, Err=5.653%.

Figura B.34: Frequenza VIII - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5.

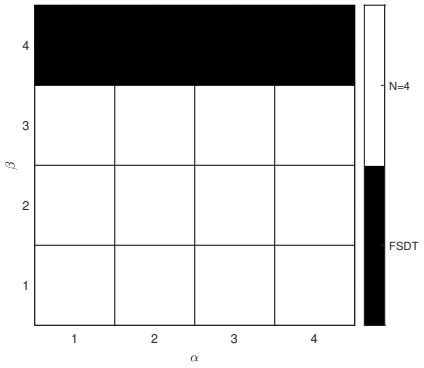


(a) FEM, Err=2.017%.

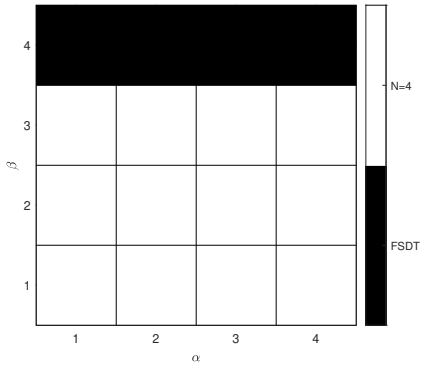


(b) NN, Err=2.108%.

Figura B.35: Frequenza VIII - Otto N4, C-F, 0/90/0, a/h=10, R/a=5.

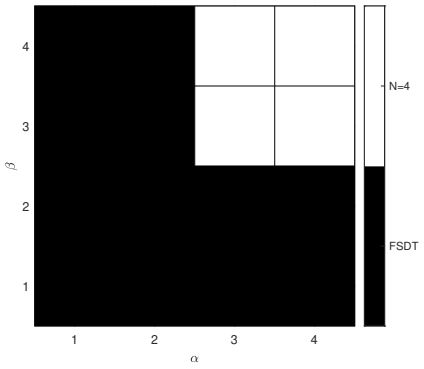


(a) FEM, Err=1.043%.

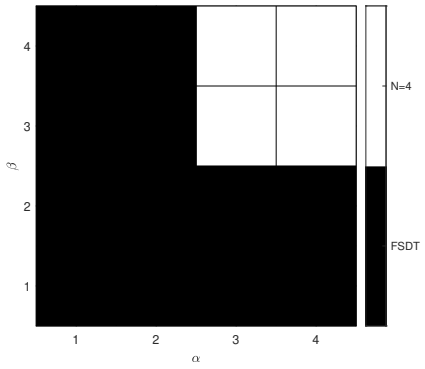


(b) NN, Err=1.009%.

Figura B.36: Frequenza VIII - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5.

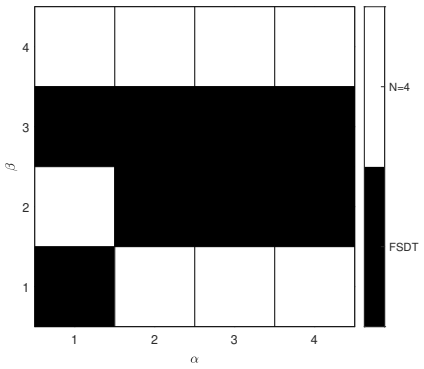


(a) FEM, Err=8.452%.

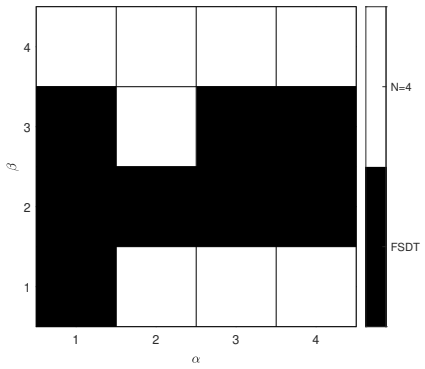


(b) NN, Err=8.358%.

Figura B.37: Frequenza IX - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5.

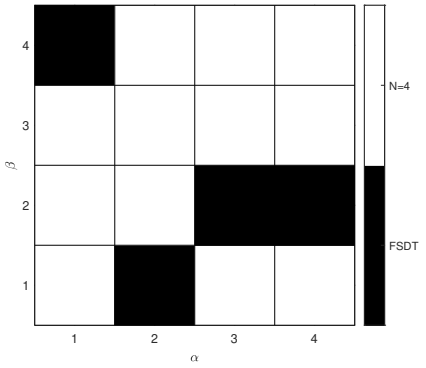


(a) FEM, Err=3.959%.

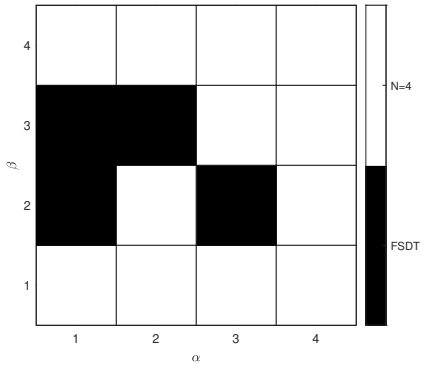


(b) NN, Err=4.264%.

Figura B.38: Frequenza IX - Otto N4, C-F, 0/90/0, a/h=10, R/a=5.

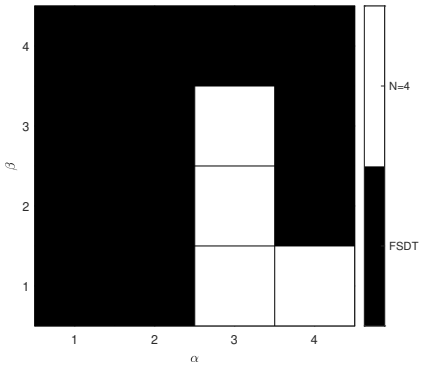


(a) FEM, Err=0.869%.

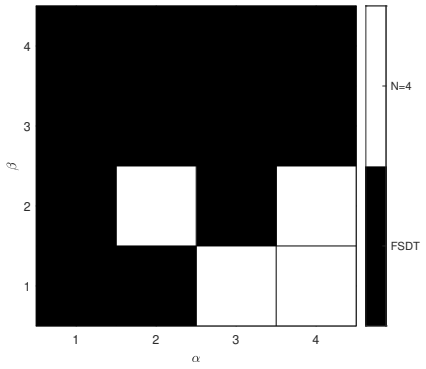


(b) NN, Err=1.245%.

Figura B.39: Frequenza IX - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5.

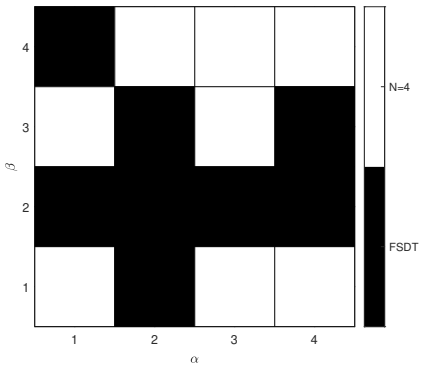


(a) FEM, Err=9.031%.

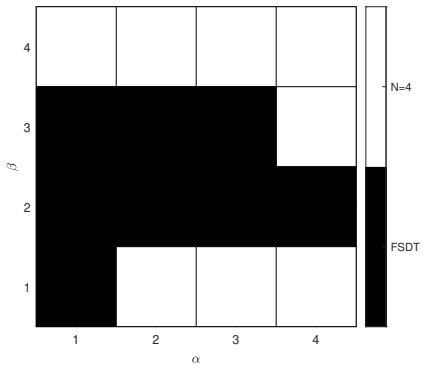


(b) NN, Err=9.621%.

Figura B.40: Frequenza X - Quattro N4, C-F, 0/90/0, a/h=10, R/a=5.

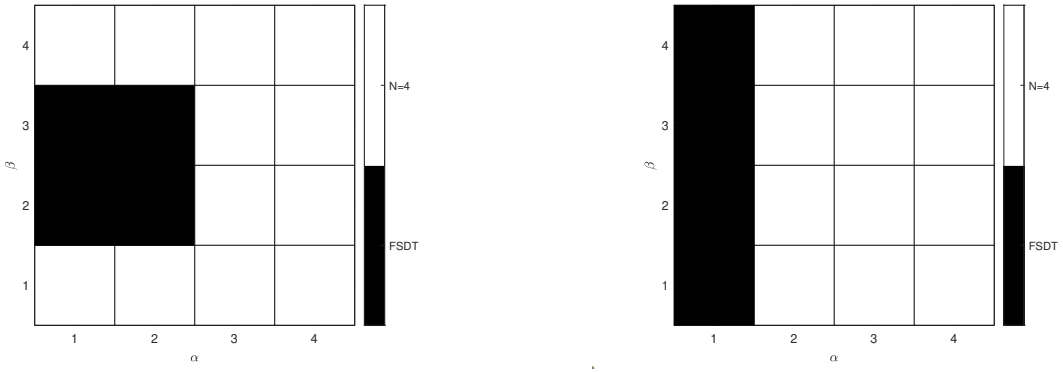


(a) FEM, Err=6.034%.



(b) NN, Err=5.856%.

Figura B.41: Frequenza X - Otto N4, C-F, 0/90/0, a/h=10, R/a=5.



(a) FEM, Err=1.514%.

(b) NN, Err=2.446%.

Figura B.42: Frequenza X - Dodici N4, C-F, 0/90/0, a/h=10, R/a=5.

Bibliografia

[1] A.L. Samuel. Some studies in machine learning using the game of checkers. *IBM Journal of Research and Development*, 3(3):210–229, 1959.

[2] Z. Ni. Reframe the field of aerospace engineering via machine learning: Application and comparison. *Journal of Physics: Conference Series*, 2386(1):012031, dec 2022.

[3] Z. Ni. Reframe the field of aerospace engineering via machine learning: Application and comparison. *Journal of Physics: Conference Series*, 2386(1):012031, dec 2022.

[4] Q. Qu, K. Liu, W. Wang, and J. Lu. Spacecraft proximity maneuvering and rendezvous with collision avoidance based on reinforcement learning. *IEEE Transactions on Aerospace and Electronic Systems*, PP:1–11, 12 2022.

[5] M. Zheng, Y. Wu, and C. Li. Reinforcement learning strategy for spacecraft attitude hyperagile tracking control with uncertainties. *Aerospace Science and Technology*, 119:107126, 2021.

[6] P.V.R. Ferreira, R. Paffenroth, A.M. Wyglinski, T.M. Hackett, S.G. Bilen, R.C. Reinhart, and D.J. Mortensen. Reinforcement learning for satellite communications: From leo to deep space operations. *IEEE Communications Society Magazine*, 57(5):70–75, May 2019.

[7] D. Ezzat, A.E. Hassanien, A. Darwish, M. Yahia, A. Ahmed, and S. Abdelghafar. Multi-objective hybrid artificial intelligence approach for fault diagnosis of aerospace systems. *IEEE Access*, 9:41717–41730, 2021.

[8] R. Gaudron and A.S. Morgans. Thermoacoustic stability prediction using classification algorithms. *Data-Centric Engineering*, 3:e17, 2022.

[9] L. Cheng, Z. Wang, and F. Jiang. Real-time control for fuel-optimal moon landing based on an interactive deep reinforcement learning algorithm. *Astrodynamics*, 2019.

[10] R. Moghe and R. Zanetti. On-line hazard detection algorithm for precision lunar landing using semantic segmentation. 01 2020.

[11] A. Pagani, M. Enea, and E. Carrera. Component-wise damage detection by neural networks and refined fes training. *Journal of Sound and Vibration*, 509:116255, 2021.

[12] C.A. Yan, R. Vescovini, and L. Dozio. A framework based on physics-informed neural networks and extreme learning for the analysis of composite structures. *Computers & Structures*, 265:106761, 2022.

[13] M. Petrolo and E. Carrera. On the use of neural networks to evaluate performances of shell models for composites. *Advanced Modeling and Simulation in Engineering Sciences*, 7, 12 2020.

[14] E. Carrera, M. Cinefra, M. Petrolo, and E. Zappino. *Finite Element Analysis of Structures Through Unified Formulation*. 07 2014.

[15] O.C. Zienkiewicz, R.L. Taylor, and J.Z. Zhu. *The Finite Element Method: Its Basis and Fundamentals, Sixth Edition*. Butterworth-Heinemann, 6 edition, 05 2005.

[16] A.S.L. Chan. Variational methods in elasticity and plasticity. k. washizu. pergamon, oxford, 1968. 350 pp. illustrated. 120s. *The Aeronautical Journal*, 72(694):889–889, 1968.

[17] K.J. Bathe. *Finite Element Procedures*. Prentice Hall, 2006.

[18] A. Pagani, E. Carrera, R. Augello, and D. Scano. Use of lagrange polynomials to build refined theories for laminated beams, plates and shells. *Composite Structures*, 276:114505, 2021.

[19] M. Petrolo and E. Carrera. Methods and guidelines for the choice of shell theories. *Acta Mechanica*, 231, 02 2020.

[20] M. Petrolo and E. Carrera. Selection of element-wise shell kinematics using neural networks. *Computers & Structures*, 244:106425, 2021.

[21] E. Grossi and M. Buscema. Introduction to artificial neural networks. *European journal of gastroenterology & hepatology*, 19:1046–54, 01 2008.

[22] W.S. McCulloch and W. Pitts. A logical calculus of the ideas immanent in nervous activity. *The Bulletin of Mathematical Biophysics*, 5(4):115–133, 1943.

[23] M.T. Hagan, H.B. Demuth, M.H. Beale, and O. De Jesús. *Neural Network Design*. Martin Hagan, 2014.

[24] A. Zhang, Z.C. Lipton, M. Li, and A.J. Smola. Dive into deep learning, 2021.

[25] J. Wu. Introduction to convolutional neural networks. 2017.

[26] A.N. Sharkawy. Principle of neural network and its main types: Review. *Journal of Advances in Applied & Computational Mathematics*, 7:8–19, 08 2020.

[27] A. Mahdavi Javid, S. Das, M. Skoglund, and S. Chatterjee. A relu dense layer to improve the performance of neural networks. 10 2020.

[28] M. Petrolo and E. Carrera. Refined beam elements with only displacement variables and plate/shell capabilities. *Meccanica*, 47:537–556, 04 2012.

[29] D.P. Kingma and J. Ba. Adam: A method for stochastic optimization, 2014.

[30] P. Nali and E. Carrera. A numerical assessment on two-dimensional failure criteria for composite layered structures. *Composites Part B: Engineering*, 43(2):280–289, 2012.

[31] Z. Hashin. Failure Criteria for Unidirectional Fiber Composites. *Journal of Applied Mechanics*, 47(2):329–334, 06 1980.

Ringraziamenti

Questa tesi rappresenta la conclusione di un percorso lungo ed impegnativo, che non sarebbe stato possibile senza il supporto di molte persone.
Ringrazio il Professore Marco Petrolo per la costante disponibilità e per avermi concesso l'opportunità di lavorare ad un argomento molto innovativo e stimolante.
Ringrazio Mattia per tutti i preziosi consigli forniti e per avermi aiutato nella prima parte pratica di questo lavoro, illustrandomi dettagliatamente come utilizzare le risorse informatiche a disposizione.
Ringrazio Pierluigi per tutto il tempo che mi ha dedicato, insegnandomi molto e permettendo confronti costruttivi su cui riflettere. Una gran parte di questo lavoro è stata possibile grazie alla sua competenza e disponibilità.
Ringrazio tutti i miei amici per le esperienze che ho potuto condividere con loro e che si sono affiancate al mio percorso universitario.
Ringrazio Anna per essermi stata vicina in questi anni, supportandomi anche in momenti difficili come le sessioni d'esami e portandomi serenità.
Un ringraziamento speciale va alla mia famiglia per aver creduto in me. Senza il loro supporto, il sostegno ed i sacrifici sostenuti, non avrei potuto raggiungere questo obiettivo.