



**Politecnico
di Torino**

Dipartimento di Ingegneria Strutturale, Edile e Geotecnica

Corso di Laurea Magistrale in Ingegneria Civile

TESI DI LAUREA MAGISTRALE

**Classificazione e tracciamento automatico dei modi
strutturali nel monitoraggio dinamico a lungo termine:**

Validazione su una chiesa barocca

Relatore:

Prof. Rosario Ceravolo

Candidato:

Rebecca Ravizza

Correlatori:

Ing. Stefania Coccimiglio

Ph.D. Erica Lenticchia

Ph.D. Gaetano Miraglia

Ing. Linda Scussolini

Anno accademico 2022/2023

*A papà, ovunque tu sia,
sono sicura che sei orgoglioso di me.*

Abstract

Lo *Structural Health Monitoring* (SHM), letteralmente monitoraggio dello stato di salute strutturale, è una branca dell'ingegneria civile strutturale che si pone come obiettivo la valutazione in continuo della condizione di strutture e infrastrutture, in modo da rilevare tempestivamente possibili variazioni o anomalie delle stesse, così da fermare in tempo l'avanzamento di eventuali stati di danneggiamento.

Nel campo dell'SHM, un'importante sottocategoria è il *Dynamic Structural Monitoring*, che si concentra sull'analisi e la creazione di dati vibrazionali.

Una delle fasi fondamentali dello SHM può essere vista nell'acquisizione di dati di accelerazione, velocità o spostamenti tramite sensori installati in situ da cui si possono ottenere informazioni fondamentali della struttura. Tali dati grezzi provenienti dagli strumenti poi vengono filtrati e analizzati in tre diversi passaggi: *Pre-processing*, fase di preparazione dei segnali raccolti, in cui vengono infatti ripuliti da rumore e fattori che disturbano i dati e che potrebbero causare valutazioni errate nei passaggi successivi; poi si passa all'*Identification*, o genericamente *Processing*, fase di effettiva analisi dei dati ottenuti, in cui l'uso di algoritmi di identificazione permette di eseguire operazioni di stabilizzazione e clustering che hanno la finalità di raggruppare i modi sulla base di frequenze e smorzamento; infine il *Post-processing* prepara i dati così ottenuti alle analisi successive.

Una volta effettuate le prime due fasi di elaborazione, il cuore di questo lavoro di tesi consiste nell'automatizzazione della fase di *post-processing* che ha visto dapprima una fase manuale e una successiva automatizzazione; ciò ha permesso di ottenere le identificazioni strutturali per l'intero periodo di cui si avevano le acquisizioni. Successivamente, tramite una procedura di *Mode Tracking* è stato possibile ottenere le serie storiche delle prime frequenze.

La procedura appena descritta è stata applicata al caso studio della Chiesa di Santa Caterina sita in Casale Monferrato (AL) esempio di spicco dell'architettura barocca religiosa piemontese e che è stata oggetto di recenti interventi di restauro e messa in sicurezza nelle parti più vulnerabili.

Indice

1	Introduzione	11
2	<i>Structural Health Monitoring</i> di beni monumentali	14
2.1	Le applicazioni dello Structural Health Monitoring	15
2.2	Il danno strutturale	16
2.3	Le fasi del monitoraggio	17
2.4	Monitoraggio dei beni architettonici	19
2.5	Tipologie di monitoraggio strutturale	19
3	Caso studio – La Chiesa di Santa Caterina	21
3.1	Inquadramento generale	21
3.1.1	Caratterizzazione storica [1]	22
3.1.2	Caratterizzazione architettonica interna	28
3.1.3	Caratterizzazione architettonica esterna	30
3.1.4	Descrizione dello schema strutturale	32
3.2	Storia del monitoraggio della chiesa	33
3.2.1	Caso studio - i sensori	34
4	Prima fase del <i>Data Processing</i> - il <i>Preprocessing</i>	36
4.1	Introduzione al <i>Data Processing</i>	36
4.2	Fase preparatoria dei dati	38
4.3	<i>Preprocessing</i>	40
4.3.1	Descrizione della procedura	41
4.4	<i>Preprocessing</i> nel caso studio	43
5	Seconda fase del <i>Data Processing</i> - l'identificazione dinamica	45
5.1	<i>Stochastic Subspace Identification</i> (SSI-CVA)	45
5.2	Descrizione del procedimento	46
5.2.1	Gestione delle variabili	46
5.2.2	Diagramma di stabilizzazione	46
5.2.3	<i>Cluster analysis</i>	47
5.3	L' <i>Identification</i> nel caso studio	48
5.3.1	Dichiarazione delle variabili	48
5.3.2	Descrizione dell'analisi	50
6	Terza fase del <i>Data Processing</i> - <i>Post-processing</i> con classificazione ma-	
	nuale	54

6.1	Ricerca dei picchi di temperatura	55
6.2	Concetti fondamentali	60
6.2.1	La stabilizzazione	60
6.2.2	Il clustering	61
6.2.3	Mediazione	61
6.2.4	I risultati	62
6.3	Classificazione manuale basata sulle frequenze e sul MAC	63
7	Automatizzazione del procedimento di <i>post-processing</i>	73
7.1	Commento al codice	73
7.1.1	<i>Post-processing</i>	75
7.1.2	<i>Post-processing_fn</i>	78
7.1.3	<i>Get_class_modes</i>	78
7.2	Salvataggio dei risultati e conclusione del processo automatico	84
7.3	Introduzione al <i>mode tracking</i>	84
8	Conclusione	86

Elenco delle figure

1	Principali passaggi per l'identificazione del danno nell'SHM	17
2	Facciata della Chiesa di Santa Caterina a Casale Monferrato (AL).	22
3	Vista della chiesa di S. Caterina in un'immagine del '900 (Associazione Santa Caterina Onlus, s.d.)	24
4	Fronte della Chiesa durante i recenti interventi di restauro.	25
5	Cronostoria della vita della Chiesa, dalla costruzioni ai tempi moderni.[2]	27
6	Planimetria principale scomposta in chiesa e coro. [3]	29
7	Interno, particolare della navata centrale.	29
8	Prospetto principale della chiesa di Santa Caterina - Facciata. [3]	30
9	Prospetto principale della chiesa di Santa Caterina - Tamburo. [3]	31
10	Prospetto principale della chiesa di Santa Caterina - Cupola. [3]	31
11	Prospetto principale della chiesa di Santa Caterina - Lanternino. [3]	32
12	Una foto tipo dei sensori utilizzati (a sx) e (a dx) il posizionamento in pianta degli accelerometri con rispettiva direzione di registrazione.	34
13	Calendario delle acquisizioni, in verde i dati utilizzabili, in rosso i periodi vacanti.	35
14	Il modello Omnibus per l'elaborazione dei dati.	36
15	Il grafico della TFD per il primo canale.	44
16	Il grafico della TFD per la media dei canali.	44
17	Le sigle delle variabili	48
18	Riassunto degli input	49
19	Dendrogramma con slot temporale di 10 minuti.	51
20	Grafico smorzamento-frequenza, con i modi candidati e quelli identificati.	52
21	Grafico ordine del sistema-frequenza, con i modi spuri e stabili.	53
22	Individuazione dei picchi di temperatura e relativi file da analizzare nel cronoprogramma delle registrazioni.	59
23	Esempio di cluster analisi in 2D.	61
24	Esempio di cluster analisi in 3D.	61
25	Grafico di stabilizzazione, con i modi candidati e quelli identificati.	63
26	Grafico ordine del sistema-frequenza, con i modi spuri e stabili.	63
27	Flow chart della logica dell'individuazione manuale dei cluster	65
28	Vettore "FRE_IDmatSort" con in evidenza i valori da scartare	66
29	Cluster validati con $\Delta = 0.4$ Hz	67
30	Cluster validati con $\Delta = 4\%$ rispetto al valore precedente	67
31	Cluster validati con $\Delta = 2.5\%$ rispetto al valore precedente	68
32	Cluster validati (in rosso) con $\Delta = 0.6$ rispetto al valore precedente, a sinistra con i cluster delle frequenze ricavati dal delta in valore assoluto, a destra con il delta in percentuale (in blu)	69

33	Cluster validati (in rosso) con $\delta = 0.8$ rispetto al valore precedente, a sinistra con i cluster delle frequenze ricavati dal δ in valore assoluto, a destra con il δ in percentuale (in blu)	69
34	Cluster validati con $MAC_i=0.945$, ottenuti dal δ di 0.8 per i cluster individuati con δ in valore assoluto della frequenza	71
35	Cluster validati con $MAC_i=0.945$, ottenuti dal δ di 0.8 per i cluster individuati con δ in percentuale	71
36	Diagramma di flusso del processo automatizzato	74
37	Esempio di un dendrogramma ricavato dall'FCM – grafico Z.	82

1 Introduzione

Gli eventi degli ultimi anni hanno permesso lo sviluppo di una maggiore consapevolezza sulla vulnerabilità delle nostre strutture, dovuta a fenomeni naturali improvvisi, come alluvioni, terremoti, ecc. . . ma anche a causa del passare del tempo.

Lo *Structural Health Monitoring* (SHM), letteralmente monitoraggio della salute strutturale, è quella branca dell'ingegneria che presenta come obiettivo la valutazione delle condizioni di salute in cui si trovano, nel caso dell'ingegneria civile, edifici e infrastrutture; in modo da rilevare tempestivamente o quasi possibili variazioni o anomalie. Tale monitoraggio permette di prevenire i danni e salvaguardare lo stato di salute delle strutture, oltre che abbassare i costi di manutenzione, tutti principi che si possono ritrovare tra gli scopi principali di questa tecnica. Le tecniche di SHM si basano sullo studio e l'analisi delle condizioni in cui una struttura si trova per scongiurare eventuali stati di danneggiamento irreversibili. Ciò implica il confronto tra due diversi stati in cui la struttura può trovarsi, lo stato iniziale assunto come non danneggiato e un secondo stato in cui la struttura risulta danneggiata, ovvero essa è in grado di soddisfare i requisiti di sicurezza ma non lavora in condizioni ottimali. Per poter studiare tali stati risulta di fondamentale importanza l'estrazione delle caratteristiche della struttura.

Nel campo dell'SHM, un'importante sottocategoria è il *Dynamic Structural Health Monitoring* (*D-SHM*) o *monitoraggio strutturale dinamico*, che si basa sull'uso di dati dinamici acquisiti utilizzando tecniche non distruttive. Questo tipo di monitoraggio permette di tenere sotto osservazione l'intera struttura in quanto fornisce informazioni di tipo globale.

Il processo di SHM prevede diverse fasi: inizia con il posizionamento di una rete di sensori, in questo caso sono stati utilizzati degli accelerometri, che permettono di acquisire dati direttamente sulla struttura. Tali dati grezzi devono essere filtrati e analizzati tramite tre diversi passaggi: per prima cosa il *Pre-processing*, fase di preparazione dei segnali raccolti, in cui vengono infatti ripuliti da rumore e frequenze non accettabili, fattori che disturbano i dati e che potrebbero causare valutazioni errate nei passaggi successivi; poi si passa alla fase di identificazione o *Identification*, o genericamente *Processing*, fase di effettiva analisi dei dati ottenuti, in cui l'uso di algoritmi di identificazione di sistemi determina l'ordine dei sistemi, poi la stabilizzazione identifica e rimuove modi instabili e infine un processo di *cluster analysis* basato su frequenze e smorzamento raggruppa i modi simili; infine il *Post-*

processing permette di preparare i dati così ottenuti alle analisi successive.

Il caso studio è rappresentato dalla Chiesa di Santa Caterina di Casale Monferrato, esempio di spicco dell'architettura barocca religiosa piemontese; essa è stata oggetto di recenti interventi di restauro e messa in sicurezza delle parti più vulnerabili della costruzione, in particolare del lanternino, zona su cui sono stati posizionati i sensori per il monitoraggio dinamico della struttura, in via temporanea – non si esclude però che questi studi possano essere le fondamenta per lo sviluppo di una rete di monitoraggio permanente su di essa.

Il cuore di questo elaborato si può ritrovare però nell'elaborazione di un processo di automatizzazione della fase di *post-processing*, utile ai fine della fase successiva di *Mode Tracking*, procedura utilizzata per ottenere le serie storiche delle frequenze.

Il presente elaborato è così diviso:

Capitolo 2: Introduzione allo *Structural Health Monitoring*, con particolare attenzione alla sua applicazione per il monitoraggio di beni architettonici. Descrizione del processo di raccolta e analisi dei dati.

Capitolo 3: Descrizione del caso studio – la Chiesa di Santa Caterina a Casale Monferrato, lo schema strutturale, la cronostoria della costruzione e gli eventi che possono aver influenzato il suo comportamento nel corso degli anni; piccola nota sui recenti lavori di ristrutturazione. Descrizione del sistema di monitoraggio, con attenzione al posizionamento dei sensori.

Capitolo 4: Descrizione del procedimento di *data processing*; spiegazione della prima fase, il *preprocessing*, con descrizione teorica della fase e l'applicazione pratica nel caso studio, con dettaglio sulla preparazione dei file e sui risultati ottenuti.

Capitolo 5: Fase di *identification*, definizione del processo, con particolare attenzione al *linkage clustering*.

Capitolo 6: Spiegazione del procedimento di *post-processing* attuato con controllo manuale, riportando parti significative del codice informatico utilizzato e descrivendo il processo logico seguito per ottenere una base da cui elaborare il processo automatico.

Capitolo 7: Descrizione del procedimento automatizzato per il *post-processing*, con *flow-chart* esplicativo del processo e commento approfondito al codice. Introduzione al *mode tracking*, per cui questo elaborato pone le basi.

Capitolo 8: Conclusione dell'elaborato, con apertura agli sviluppi futuri.

2 *Structural Health Monitoring* di beni monumentali

Il procedimento che riguarda la ricerca della strategia di identificazione dei danni nel campo dell'ingegneria aerospaziale, civile e meccanica prende il nome di *Structural Health Monitoring*, in italiano monitoraggio della salute strutturale.

Esso consiste nell'osservazione di una struttura per un determinato periodo di tempo, attuando misurazioni periodiche, al fine di individuare debolezze, sensibili quindi a danni, e anche a volte analisi statistiche che permettano la valutazione della condizione attuale del sistema stesso.

L'SHM ha ricevuto grande attenzione e ha visto il suo sviluppo maggiore attuarsi negli ultimi 40 anni, in particolare l'applicazione nel campo dell'ingegneria civile è stata studiata tramite la valutazione dei danni strutturali a partire da dati vibrazionali di ponti e strutture dagli anni '80. Le difficoltà principali che questa tecnica affronta nel campo sono la variabilità delle condizioni atmosferiche e ambientali, insieme alle dimensioni solitamente grandi delle strutture da monitorare. [4]

I casi per cui può essere utile il monitoraggio strutturale sono stati identificati da Ross e Matthews (1995)[5] e da Mita (1999)[6] :

- Modifiche a strutture già esistenti;
- Strutture soggette a lavori esterni;
- Demolizioni;
- Strutture soggette a spostamenti a lungo termine o deterioramento dei materiali;
- Ciclo di feedback per migliorare progetti futuri basandosi sull'esperienza;
- Valutazione della fatica;
- Nuovi sistemi di costruzione;
- Valutazione dell'integrità strutturale in seguito ad attività sismica;
- Decadenza della costruzione e crescita della necessità di manutenzione;

- Propensione ad una filosofia di costruzione basata sulle performance attese.

2.1 Le applicazioni dello Structural Health Monitoring

Sviluppi significativi di questa tecnica si sono originati grazie all'applicazione su progetti strutturali importanti, come dighe, ponti di grande luce e piattaforme off-shore a partire dagli anni '80.[7]

I primi esempi che ne vedono l'applicazione pratica infatti sono proprio le dighe, dal momento che in Inghilterra sin dalla metà del XIX secolo era per legge obbligatoria l'ispezione di tali infrastrutture, e a partire dal 1975, con il Reservoir Act, fu assegnata la responsabilità non solo della supervisione, ma anche dell'interpretazione dei dati operativi delle dighe alla figura dell'ingegnere.[7] Il monitoraggio della risposta strutturale delle dighe è di duplice importanza, sia per la valutazione della sicurezza nei confronti di eventi sismici, sia per la valutazione della condizione strutturale per garantire l'operatività ottimale della struttura.

Un'altra importante nota va all'applicazione dell'SHM sui ponti di grande luce, poiché rappresentano un ideale campo di prova per l'implementazione del sistema, più che per un'effettiva valutazione strutturale. Ciò avviene perché nelle strutture così imponenti, sollecitate per lo più da carichi dinamici come il vento, i programmi di ispezione e manutenzione portano alla detezione precoce di eventuali difetti o danni anche visivamente, mentre al contrario con i sistemi di monitoraggio sarebbe necessario posizionare una rete molto densa di sensori, con una notevole spesa economica – i vantaggi però che offrono all'implementazione e al miglioramento della tecnica dell'SHM sono tali da permetterne l'applicazione.

È interessante il caso delle piattaforme off-shore, strutture il cui monitoraggio, reso obbligatorio a partire dagli anni '70, rappresentava una sfida, sia economica sia per il pericolo che ne comportava, essendo necessariamente da attuare con un tecnico in subacquea. Si era pensato quindi che l'applicazione del monitoraggio dinamico a questa tipologia di strutture potesse risolvere tali problemi, ma questa idea si è rivelata errata per la non-stazionarietà del sistema e per i continui cambiamenti del sistema, dati per esempio dal movimento delle operazioni effettuate su di esse.[7]

Un ulteriore esempio può essere quello che riguarda strutture ad uso civile, come torri o grattacieli, per cui inizialmente il monitoraggio dinamico si era proposto come analisi delle condizioni strutturali in caso di eventi sismici, ma i suoi campi applicativi si sono poi allargati per comprendere i cicli di carico e scarico, come per il vento. Attualmente l'uso più

comune che si fa di questa tecnica sulle strutture è l'osservazione dei parametri dinamici, quali frequenze naturali e forme modali.

2.2 Il danno strutturale

È indispensabile introdurre il concetto di danno che può essere definito come un cambiamento nel sistema che influisce negativamente sulle sue condizioni presenti o future, limitato non solo quindi a livello dei materiali o della geometria, ma che si può estendere anche alle condizioni al contorno, qualora inficino le performance della struttura.

Naturalmente ciò implica la necessità di comparazione tra due stati, l'uno iniziale e il secondo danneggiato, da cui si pone l'interrogativo della ricerca di tali stadi, in particolare di quest'ultimo. [4]

Worden e Dulieu-Barton [8] hanno definito diversi livelli di danno, concatenati tra loro per gravità crescente, nel seguente modo:

- Un *defect* (traducibile come difetto), inerente al materiale e statisticamente contenuto in ognuno in quantità variabile, che quindi non influenza l'operatività a livello globale del sistema;
- Il *damage* (danno), applicabile quando il sistema non si presenta più nelle condizioni ottimali, ma è ancora in condizioni soddisfacenti di operatività;
- Un *fault* (guasto), ovvero il cambiamento nel sistema che produce una riduzione non accettabile della qualità del sistema, dove con qualità si intende la sua condizione di funzionamento ottimale; questo per Farrar [4] è invece definito come stato di *failure* (collasso).

Si evince quindi la difficoltà nello stabilire il raggiungimento di uno di questi stadi per ogni sistema, poiché risultano essere dipendenti non solo dalle caratteristiche dello stesso, quindi i materiali e lo schema strutturale, ma sono anche fortemente influenzati da altri fattori, quali le condizioni ambientali o la variabile temporale, sia a lungo termine, fenomeni di fatica, sia a breve termine, scosse di terremoto.

Per l'identificazione intelligente del danno strutturale è necessario seguire una serie gerarchica di passaggi, individuati prima da Rytter (1993) [9] e perfezionati da Worden e Dulieu-Barton[8]:

1. Rilevamento, dove viene fornita l'indicazione che potrebbe essere presente un danno nella struttura;

2. Localizzazione, la fase che individua il possibile posizionamento di tale danno;
3. Classificazione, ricerca della tipologia del danno – passaggio fondamentale;
4. Valutazione, stima dell'estensione e della gravità del danno;
5. Predizione, metodo che individua informazioni riguardo la sicurezza e la vita residua della struttura.



Figura 1: Principali passaggi per l'identificazione del danno nell'SHM

2.3 Le fasi del monitoraggio

Al fine di descrivere il processo di applicazione dell'SHM, è ottimale la divisione in quattro passaggi del *statistical pattern recognition paradigm approach*, per i metodi *data-driven*, come quello descritto in questo elaborato.

Con il metodo *data-driven*, si individua lo stato di danno tramite il confronto dei dati raccolti, implementati con algoritmi di identificazione del modello, con il modello di riferimento corrispondente - finchè i due comportamenti sono simili, si può escludere il danno sulla struttura, se si discostano significa che è stata rilevato un danneggiamento.

Questo approccio si contrappone ai metodi invece *model-driven*, in cui si procede secondo un problema inverso, che prevede prima con la creazione di un modello ad alta fedeltà della struttura, con cui si possono confrontare i dati del monitoraggio.[10]

Le fasi del procedimento *data-driven* possono essere sintetizzate come segue:

- **Valutazione operativa** – in questo passaggio è necessario effettuare una sorta di analisi costi-benefici, individuando i vantaggi in termini di sicurezza o vita della struttura in relazione alla giustificazione economica della sua applicazione. È importante anche definire la condizione, o la possibilità che si verifichi, di danno sulla struttura interessata, e le condizioni operative e ambientali in cui il sistema di monitoraggio verrebbe applicato, con eventuali limitazioni nell'acquisizione dei dati, vedasi il caso delle piattaforme off-shore di cui sopra.
- **Acquisizione dei dati, normalizzazione e pulizia** – mentre per l'acquisizione dei dati è fondamentale la scelta del tipo di sensori, del loro posizionamento e degli hardware di registrazione, la normalizzazione assume un ruolo importante poi per la separazione delle informazioni registrate, se imputabili al danno strutturale o se semplicemente frutto della variazione delle condizioni operative o ambientali. Il passaggio quindi della pulizia è attuato con il fine di ottenere acquisizioni consistenti, scegliendo quali dati accettare e quali scartare.
- **Estrazione delle caratteristiche e condensazione delle informazioni** – è la fase che più richiede attenzione nel processo, perché il suo scopo è quello di distinguere tra la condizione danneggiata e la condizione pre-danneggiamento. Gli strumenti che si utilizzano in questa fase sono ad esempio i modelli FEM, in cui viene simulato il danno al fine di attuare poi una serie di esperimenti numerici. È importante sottolineare che la raccolta dei dati si attua su periodi di tempo molto lunghi ed è quindi fondamentale l'applicazione di corrette tecniche di riduzione dei dati, per ottenere solo informazioni robuste.
- **Sviluppo del modello statistico** – è la fase che riguarda l'implementazione dell'algoritmo di estrazione delle feature caratterizzanti lo stato danneggiato della struttura. Possono classificarsi in due gruppi, il *supervised learning*, in cui sono accessibili dati sia sulla struttura non danneggiata che quella soggetta a danno, o *unsupervised learning*, quando non si hanno a disposizione i dati sulla struttura danneggiata. Un altro scopo di questo passaggio è l'individuazione delle false indicazioni di danni, i falsi positivi (danno individuato quando non presente) o falsi negativi (danno non individuato quando presente).[4]

2.4 Monitoraggio dei beni architettonici

Il nostro paese, l'Italia, noto nel mondo per la ricchezza e la varietà del suo patrimonio storico-monumentale, la cui conservazione dovrebbe risultare tra gli obiettivi più importanti, può ritrovare nella tecnica dello SHM un grande alleato. Infatti nel campo della conservazione e del restauro dei beni architettonici, proprio come la chiesa, oggetto del presente documento, lo SHM riveste un ruolo fondamentale nella preservazione degli edifici storici. Questa tecnologia avanzata consente di monitorare costantemente la condizione strutturale dell'edificio, fornendo dati preziosi ai restauratori e ai conservatori.

Uno dei primi esempi di applicazioni di SHM nel paese si può trovare infatti in uno dei monumenti più rappresentativi del panorama architettonico italiano, il Duomo di Santa Maria in Fiore a Firenze; il sistema è già attivo dal 1988 ed è formato da più di 160 sensori statici, necessari a monitorare la complessità dello stato fessurativo in cui si trova la struttura.[10]

I beni architettonici, soprattutto in muratura, presentano come sfida aggiuntiva la loro complessità intrinseca, data dall'incertezza della conoscenza del loro stato di salute strutturale, non solo per le trasformazioni che hanno subito nel tempo, ma anche per i possibili danneggiamenti non visibili presenti.

Il *Vibration-based SHM* basa il suo principio di applicazione sul concetto che si possano individuare stati di danno nelle strutture tramite il modo in cui essa risponde a sollecitazioni di tipo vibrazionale registrabili con strumentazione non invasiva; le informazioni che si ottengono in questo modo, permettono di ottenere anche dati, come le frequenze, che con altri metodi nelle stesse condizioni non si riuscirebbero a ricavare nei casi in cui massa e rigidità della struttura sotto monitoraggio non fossero note.

Il suo principale punto di forza su questo tipo di strutture è l'ottenimento della situazione globale della struttura e la valutazione del comportamento della stessa, in maniera non invasiva e non distruttiva – caratteristica fondamentale nei casi in cui sia preferibile la conservazione dei materiali nella ricerca di informazioni sul comportamento strutturale.

2.5 Tipologie di monitoraggio strutturale

Con il termine monitoraggio strutturale si possono definire molte tecniche di analisi delle condizioni della struttura e quindi ne deriva la necessità di dare una iniziale divisione a seconda della strumentazione impiegata. Infatti si può già definire un'iniziale distinzione tra il monitoraggio visivo e quello strumentale, dove il primo rappresenta il punto di partenza

per il secondo.

L'ispezione visiva, che può essere considerata come una tipologia di monitoraggio, rappresenta il controllo periodico per la ricerca di fenomeni di danno, di degrado, stati fessurativi; per informazioni più approfondite però sullo stato di danno, anche non visibile, si rende necessaria però un'analisi con supporto strumentale.

Il monitoraggio di tipo strumentale a sua volta si divide in due categorie:

- Monitoraggio statico, che si basa sulla misurazione di variabili come pressione, temperatura o umidità, tramite apposta strumentazione, su lunghi periodi con intervalli di tempo distanziati – questo tipo di informazioni vanno poi associate a una conoscenza approfondita della struttura, senza dimenticare l'accurata analisi dei software preposti dei dati raccolti.
- Monitoraggio dinamico, che invece ha lo scopo di fornire la risposta sismica o la caratterizzazione dinamica della struttura.

Quest'ultimo può essere applicato in diverse modalità temporali, infatti può essere necessaria l'installazione di un sistema *ad hoc*, ovvero in una sola occasione e per un periodo di tempo ristretto, o di un sistema a lungo termine, come quello in esame nel presente documento, quindi su un periodo più lungo, per arrivare ai sistemi permanenti, che richiedono però una grande possibilità di archiviazione dei dati, se registrati in continuo, oppure possono essere del tipo *trigger-based*, ovvero attivati quando determinati valori superino una determinata soglia.[11]

Le tipologie di variabili registrate da questo tipo di monitoraggio sono parametri come spostamenti, velocità o accelerazioni, e sono utili non solo alla conoscenza in tempo reale dello stato della struttura e quindi per la detezione del danno precoce, ma anche per il controllo della risposta ad eventi eccezionali di tipo dinamico, come ad esempio un sisma.

3 Caso studio – La Chiesa di Santa Caterina

3.1 Inquadramento generale

La Chiesa di Santa Caterina, che sorge nel centro storico di Casale Monferrato, fu edificata da Giacomo Zanetti su progetto di Giovanni Battista Scapitta tra il 1718 e il 1726 ed è senza dubbio una delle più importanti manifestazioni del barocco piemontese, risultando essere la migliore espressione dell'architettura sacra settecentesca della città.

La pianta presenta una disposizione a croce greca, dominata dalla cupola ellittica costolata, completamente affrescata, imposta sul tamburo finestrato a otto spicchi. La cupola è sormontata da un piccolo lanternino a pianta ellittica, che viene sorretto da sottili pilastri.

Il bene d'interesse storico e artistico è tutelato ai sensi del Decreto Legislativo 22 gennaio 2004, n. 42 "Codice dei beni culturali e del paesaggio".

Si procederà ora nel presente capitolo con un'analisi storico critica, in modo da inquadrare le diverse fasi della costruzione dell'edificio ripercorrendone la storia, passaggio fondamentale per progettare interventi di miglioramento o per comprendere meglio il comportamento a sollecitazione della struttura, ed è proprio questo secondo aspetto la colonna portante del presente documento.

Verranno anche presi in considerazione le modifiche strutturali a cui è stata soggetta, oltre agli eventi che possono aver influito su di essa, in modo da comporre un quadro quanto più completo possibile sull'edificio. In particolar modo, si porrà attenzione a eventuali eventi di tipo sismico, registrati nel sito in analisi o nelle zone limitrofe, e che quindi possono essere stati impattanti, in modo più o meno incisivo, sulla struttura.

È da notare che saranno considerati anche i recenti interventi di miglioramento e riparazione di parti della chiesa, che comportano una variazione del livello di conoscenza della struttura.



Figura 2: Facciata della Chiesa di Santa Caterina a Casale Monferrato (AL).

3.1.1 Caratterizzazione storica [1]

Le indicazioni delle norme attualmente vigenti (NTC 2018) riportano che “ai fini di una corretta individuazione del sistema strutturale esistente e del suo stato di sollecitazione è importante ricostruire il processo di realizzazione e le successive modificazioni subite nel tempo dal manufatto, nonché gli eventi che lo hanno interessato”.

Ciò verrà attuato tramite il reperimento di dati e informazioni, grazie a indagini archivistiche, per ricostruire i più rilevanti passaggi storici, ma verrà anche effettuata un’analisi degli aspetti strutturali e della composizione architettonica, che permetteranno l’individuazione degli elementi principali e caratteristici che la compongono.

La chiesa, ad oggi nota come Santa Caterina, ufficialmente intitolata a Santa Maria delle Grazie, si trova nel centro cittadino di Casale Monferrato, in Piazza Castello, delimitata nel lato sud dal centro storico della città. La chiesa è stata commissionata come luogo di culto a completamento del convento delle Monache Domenicane, dedite a Santa Caterina da Siena, da cui appunto deriva il nome. La storiografia tradizionale attribuisce la paternità del progetto all'architetto e agrimensore del Ducato del Monferrato, Giovanni Battista Scapitta, ma le notizie successive hanno smentito questa notizia, principalmente per motivi cronologici, senza fornire un'attribuzione alternativa. Attualmente quindi, di fatto, non esiste un progettista certo, ma è possibile assegnare a Donato Zanetti il ruolo di capomastro in base alla scoperta di un documento presso l'Archivio Capitolare di Casale relativo a un intervento effettuato per risolvere un problema strutturale nella chiesa.

L'aspetto attuale della chiesa rappresenta una delle interpretazioni più rappresentative e significative del barocco piemontese, frutto di una serie di interventi e ampliamenti che si sono svolti nel corso dei secoli. La sua storia ha inizio nel 1528, quando la Marchesa Anne D'Alençon, in qualità di amministratrice dei beni del figlio, il Marchese del Monferrato, dona alle monache domenicane il palazzo marchionale (Palazzo Trevisio), che era stato recentemente ristrutturato dalla stessa marchesa, con l'obiettivo di creare un luogo di preghiera per le monache.

La chiesa originaria, risalente al XVI secolo, fu ottenuta utilizzando il piano terra dell'ala nord del palazzo marchionale, che si affacciava su via Trevigi. Essa era composta da un'aula interna a uso esclusivo delle monache, il Coro delle Monache, collegato direttamente al nucleo centrale del convento (Palazzo Trevisio) e da un'aula esterna riservata ai fedeli, che si affacciava su Piazza Castello. Nel corso del XVII secolo, il luogo di culto subì un ampliamento verso piazza Castello, con la realizzazione di una nuova chiesa che rispettava le regole architettoniche imposte dalle normative dei conventi di clausura stabilite dal Concilio di Trento.

Nel primo periodo del XVIII secolo, la chiesa subì ulteriori trasformazioni, pur mantenendo la suddivisione in due aule; queste modifiche conferirono alla chiesa un aspetto più articolato e in linea con i gusti architettonici e decorativi del periodo barocco e le conferirono un aspetto caratterizzato da forme morbide e sinuose, ricchi elementi decorativi e stucchi sontuosi, già presenti in facciata.

Mentre il Coro attuale, identificato come la chiesa interna, fu completato nel 1715, la chiesa attualmente adibita all'uso pubblico, con vista su Piazza Castello e nota come chiesa esterna, fu costruita a partire dal 1718 e terminata nel 1726.

L'apparato decorativo, che include finiture pregiate in pietra e marmo e numerosi affreschi all'interno del tempio, fu realizzato in un periodo abbastanza esteso, completato solo



Figura 3: Vista della chiesa di S. Caterina in un'immagine del '900 (Associazione Santa Caterina Onlus, s.d.)

nella metà del XVIII secolo. Anche per quanto riguarda gli affreschi, il riconoscimento dei pittori Gian Carlo Aliberti, il Cavaliere Benaschi e Vittore è ancora in fase di studio. Le ricerche negli archivi testimoniano anche diversi interventi che hanno coinvolto nuovamente la chiesa nel XIX e XX secolo. Questi interventi hanno principalmente mirato a mantenere la consolidata conformazione architettonica dell'edificio, con opere incentrate principalmente sulla conservazione e il restauro della chiesa.

Al fine di fornire un quadro storico chiaro e dettagliato che copra le diverse vicissitudini che hanno coinvolto la chiesa di Santa Caterina, dalla sua origine fino ai giorni nostri, vengono riassunte di seguito le informazioni ottenute dalle ricerche di archivio condotte dall'Associazione Santa Caterina Onlus di Casale Monferrato. Questa ricostruzione degli eventi è fondamentale per aumentare la conoscenza di questo patrimonio protetto nel contesto del presente lavoro.

La Direttiva BBCC del 26 febbraio 2011, nel paragrafo 4.1.5, sottolinea l'importanza di ricostruire l'intera storia costruttiva del bene culturale tutelato al fine di identificare correttamente il suo sistema strutturale e il suo stato di sollecitazione. È essenziale evidenziare la successione dei vari interventi di costruzione e le successive modifiche nel tempo del manufatto, al fine di individuare eventuali zone di discontinuità e disomogeneità sia in pianta che in alzato, come aggiunte, sopraelevazioni, sostituzioni di parti strutturali, e così via. Nello specifico afferma: "Per una corretta valutazione della stabilità strutturale e delle sollecitazioni che agiscono sul bene culturale tutelato, è fondamentale ricostruire l'intera storia costruttiva dell'edificio, compresi i processi di costruzione e le successive modifiche nel corso

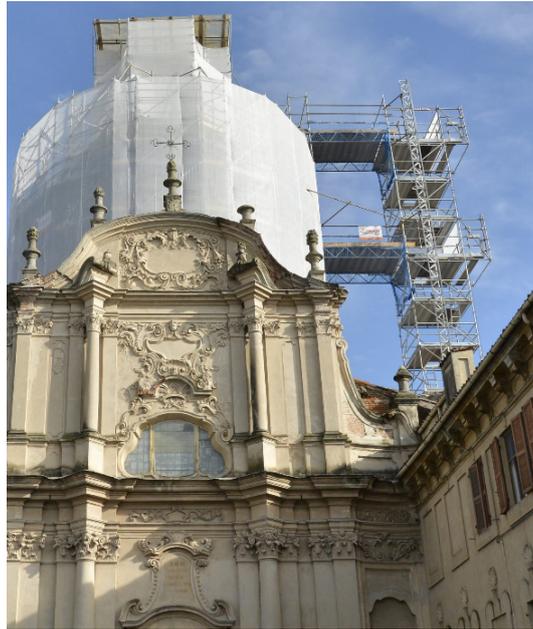


Figura 4: Fronte della Chiesa durante i recenti interventi di restauro.

del tempo. In particolare, occorre evidenziare la successione cronologica delle diverse parti dell'edificio al fine di individuare eventuali zone di disconnessione o materiali diversi, sia in pianta che in alzato (come aggiunte, sopraelevazioni, sostituzioni di elementi architettonici, ecc.)”.

In particolare, nella tabella vengono evidenziati gli interventi significativi che hanno coinvolto la chiesa in termini di manutenzione, ripristino, restauro e consolidamento strutturale. Queste testimonianze del passato consentiranno di ricostruire la storia dell'opera e fornire un quadro completo per aumentare la conoscenza della stessa.

Al fine di tracciare una cronologia dettagliata e completa delle vicissitudini che hanno coinvolto la chiesa di Santa Caterina, dalla sua fondazione fino ai giorni odierni, abbiamo riassunto la sua storia basandoci sulle ricerche archivistiche condotte dall'associazione Santa Caterina Onlus di Casale Monferrato. La ricostruzione degli eventi si rivelerà preziosa per aumentare la conoscenza di questo patrimonio tutelato. Queste testimonianze del passato ci permettono di ricostruire la storia dell'edificio e di ottenere una visione completa per una migliore comprensione dello stesso.

Anno	Evento	Descrizione
1528	Fondazione della chiesa	La Marchesa Anne D'Alençon dona il palazzo marchionale alle monache domenicane di Santa Caterina da Siena.
1560	Cambio di intitolazione	Papa Pio IV decreta che la chiesa sia intitolata a Santa Maria delle Grazie anziché a Santa Caterina.
1718	Inizio costruzione della nuova chiesa	Viene posta la prima pietra per la costruzione della nuova chiesa, rivolta verso la piazza.
1720	Lavori sulla chiesa esterna	Si iniziano i lavori sulla chiesa esterna con la supervisione di Giacomo Zanetti come Capo Mastro da muro.
1723	Ultimazione della nuova chiesa	La nuova chiesa è quasi completata, ma le monache sono indebitate e si segnalano danni alle vetrate della cupola.
1724	Completamento della chiesa interna	La costruzione e decorazione della chiesa interna dedicata alla Santa Croce sono terminate.
1726	Consacrazione della chiesa esterna	La chiesa esterna viene consacrata da Mons. Radicati alla presenza del Padre Guardiano e del Confessore dell'Ordine dei Predicatori.
1780	Benedizione del gruppo marmoreo d'altare	Il gruppo marmoreo d'altare, realizzato dallo scultore Giovanni Battista Bernero, viene benedetto.
1802	Soppressione degli ordini religiosi	Gli ordini religiosi vengono soppressi e i beni delle monache vengono confiscati.
1814	Riapertura della chiesa	Il Re Vittorio Emanuele I istituisce il Reale Collegio di Educazione a Casale e riapre la chiesa al culto.
1866	Lavori di restauro	Vengono eseguiti importanti lavori di restauro su cupola, cupolino e tetti della chiesa.
1901, 1911	Lavori di manutenzione	Vengono effettuati lavori di manutenzione sulle coperture.
1927	Sollecito per manutenzioni	La Soprintendenza richiede all'Ente Treviso di provvedere alle urgenti manutenzioni della chiesa.
1931	Ripresa delle attività dei Somaschi	I Padri Somaschi riprendono le attività al Collegio Treviso e ottengono il permesso di officiare nella chiesa.

1940	Restauro della facciata	La Soprintendenza sollecita l'Ente Trevisio a effettuare il restauro della facciata.
1943	Importanti lavori di restauro	Vengono realizzati importanti lavori di restauro su cupole, coperture e facciata.
1968	Inizio dei restauri	I Padri Somaschi avviano i lavori di restauro e installano l'impianto di riscaldamento.
1969	Riapertura al pubblico	Dopo due anni di lavori interni ed esterni, la chiesa riapre al pubblico sotto l'Intendenza delle Belle Arti.
1973	Partenza dei Padri Somaschi	L'ordine dei Padri Somaschi lascia definitivamente Casale, e la chiesa viene affidata alla Diocesi.
1980	Restauro della facciata	L'Ente Trevisio richiede l'autorizzazione per il restauro della facciata e degli ornamenti in stucco di calce.
1993	Morte di Giuseppe Zavattaro	Muore Giuseppe Zavattaro, diacono e custode della chiesa per tutta la vita.
2002	Lavori di manutenzione sui tetti	La Soprintendenza autorizza lavori di manutenzione straordinaria sui tetti.
2010	Fondazione dell'Associazione Santa Caterina Onlus	Viene fondata l'Associazione per la valorizzazione e il restauro della chiesa di Santa Caterina.
2011	Riapertura al pubblico	La chiesa viene riaperta al pubblico dall'Associazione Santa Caterina Onlus, e il tetto del coro viene restaurato.
2019	Inizio dei lavori di restauro	Viene inaugurato il cantiere di restauro di lanterna, cupole e tamburo.
2021	Completamento dei lavori di restauro	Vengono conclusi i lavori su lanterna, cupole, tamburo e tetti a falde. Si attua rinforzo della facciata.

Figura 5: Cronostoria della vita della Chiesa, dalla costruzioni ai tempi moderni.[2]

3.1.2 Caratterizzazione architettonica interna

Nonostante i numerosi secoli di storia che l'hanno attraversata, la chiesa di Santa Caterina mantiene ancora oggi la magnificenza e l'imponenza conferitele all'inizio del Settecento, durante l'epoca del barocco. La sua architettura riflette appieno gli stilemi di questa corrente artistica, evidenti nelle forme sinuose della facciata, negli elaborati decori e nella maestosa cupola ellittica che sovrasta l'intera struttura.

Da un punto di vista architettonico, la chiesa presenta una pianta a croce greca, con un'ampia aula centrale di ingresso sormontata da una maestosa cupola ellittica. Quest'ultima si appoggia su un tamburo finestrato diviso in otto spicchi e è coronata da una lanterna composta da otto pilastri intervallati da vetrate. La copertura della cupola e della lanterna è realizzata in metallo nervato, con decorazioni a forma di pigna in pietra e metallo. Il cupolino è coronato da un grande fastigio a forma di globo e croce.

All'interno della chiesa si trova un'ampia navata centrale, affiancata da matronei, e delimitata lateralmente da due cappelle completamente simmetriche, entrambe con volte affrescate sorrette da pilastri laterali. La cupola, vista dall'interno, presenta preziosi affreschi e dorature, così come il tamburo su cui si appoggia. Il tamburo è caratterizzato da coppie di colonne in finto marmo, separate da otto finestre polilobate. A sua volta, il tamburo si basa su quattro volte a cassettoni sorrette da quattro pilastri principali.

Alla fine della navata centrale si trova il presbiterio, che termina con un'abside di forma semicircolare. Anche il presbiterio è sormontato da una cupola di dimensioni più piccole rispetto a quella centrale, con pianta circolare. La parete retrostante al presbiterio separa la cosiddetta "Chiesa Esterna", o chiesa principale dove si svolgevano le messe pubbliche, dalla "Chiesa Interna", o coro, dove le suore claustrali assistevano alle funzioni liturgiche. Questa parete serviva a delimitare lo spazio di chiusura ed è ancora presente un'apertura posta in alto, con inferriate in ferro battuto.

Il coro, che si trova dietro la chiesa, ha una pianta rettangolare con dimensioni di circa 10 metri di larghezza e 22 metri di lunghezza. Lungo il suo perimetro si trova un coro ligneo, adornato da stalli intagliati con sedili ribaltabili e colonnine slanciate con capitelli corinzi. Nella parte terminale del coro c'è un'ampia semicupola incorniciata, al centro della quale si trova una cantoria con decorazioni dorate e una balconata dipinta in legno. Le pareti interne del coro, ad eccezione di quella che confina con la chiesa, sono decorate con paraste e archi a tutto sesto sormontati da catene metalliche leggermente sporgenti rispetto alla volta a botte.

Vi è inoltre un loggiato adiacente sia alla chiesa che al coro, che si affaccia sul cortile della scuola Trevigi, composto da una serie di archi e colonne e si sviluppa su tre piani. Al piano terra e al primo piano si trovano volte a crociera sostenute da pilastri in muratura,

mentre all'ultimo piano c'è una copertura in legno.

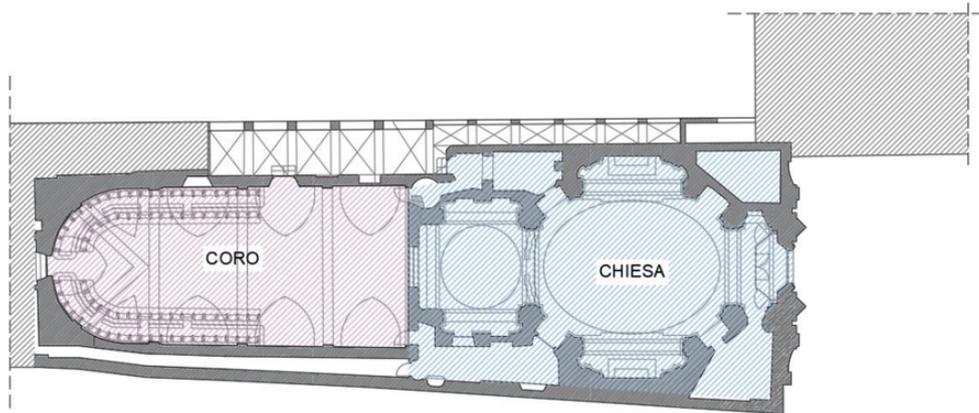


Figura 6: Planimetria principale scomposta in chiesa e coro. [3]



Figura 7: Interno, particolare della navata centrale.

3.1.3 Caratterizzazione architettonica esterna

Dal punto di vista esterno, tra gli elementi architettonici che caratterizzano maggiormente la chiesa c'è la facciata principale, che si estende per circa 19 metri e sporge rispetto alla struttura principale dal livello corrispondente alla base del tamburo, che si trova a circa 13 metri di altezza. In alto, è decorata con un notevole e ricercato frontone dalle linee mistilinee, che prosegue per circa 6 metri.

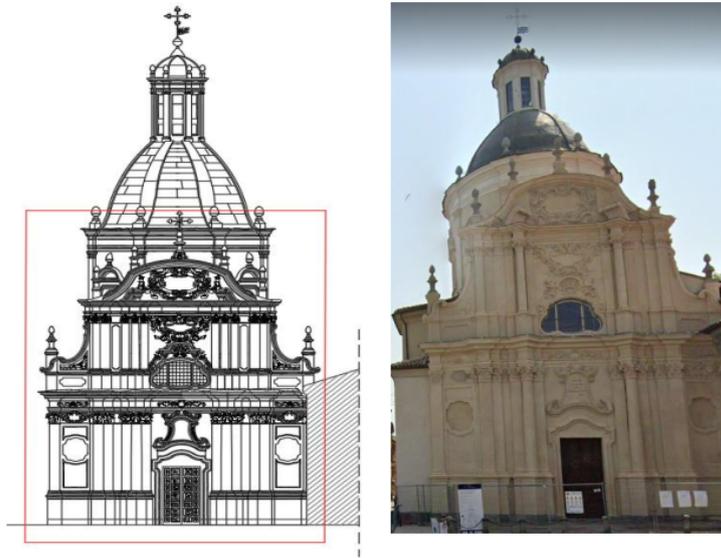


Figura 8: Prospetto principale della chiesa di Santa Caterina - Facciata. [3]

Il tamburo poi è posizionato a partire da un'altezza di circa 13 metri e si estende per circa 7 metri. La sua pianta è ellittica, con un asse maggiore di circa 15 metri e un asse minore di circa 10 metri. La superficie esterna del tamburo è caratterizzata da una regolare successione di otto paraste che sostengono la cupola e la lanterna.

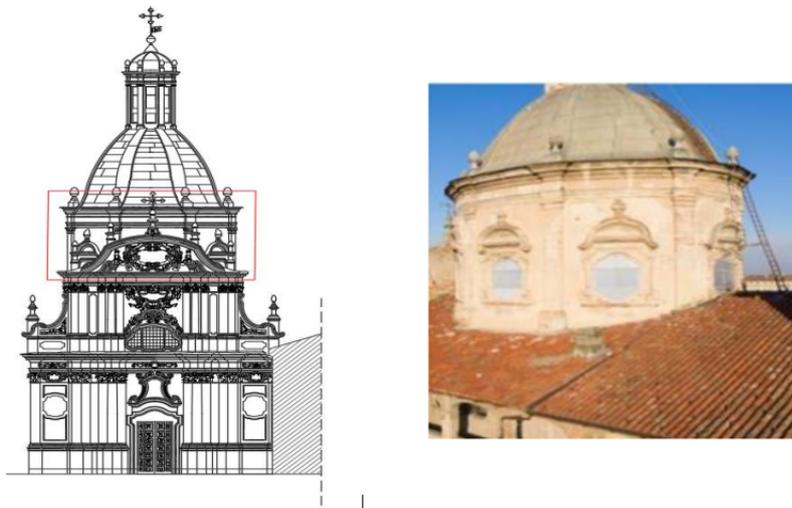


Figura 9: Prospetto principale della chiesa di Santa Caterina - Tamburo. [3]

La cupola è collocata sul tamburo, formando un unico sistema con esso. Si trova ad un'altezza di circa 21 metri da cui si innalza per 4,50 metri. Dal punto di vista strutturale, presenta otto costoloni che sporgono sia nella parte esterna che in quella interna, e sono sostenuti da apposite strutture murarie; la sua copertura è realizzata con lastre di piombo.



Figura 10: Prospetto principale della chiesa di Santa Caterina - Cupola. [3]

La lanterna si trova sulla cupola a 26,50 metri ed è alta 6 metri (escludendo il "pennacchio" e la croce). La superficie esterna della lanterna è caratterizzata da otto colonnine, con una sezione fortemente irregolare ma che può essere ricondotta ad una forma rettangolare. Queste colonnine sono posizionate su un piedistallo costituito da un semplice dado, senza

zoccolatura né cimasa. La lanterna è la base su cui sono stati posizionati gli strumenti per l'analisi condotta in questo documento, ovvero gli accelerometri, che appunto in virtù della grande altezza della loro posizione, hanno potuto rilevare gli spostamenti (tramite l'accelerazione) della struttura nelle due direzioni principali.

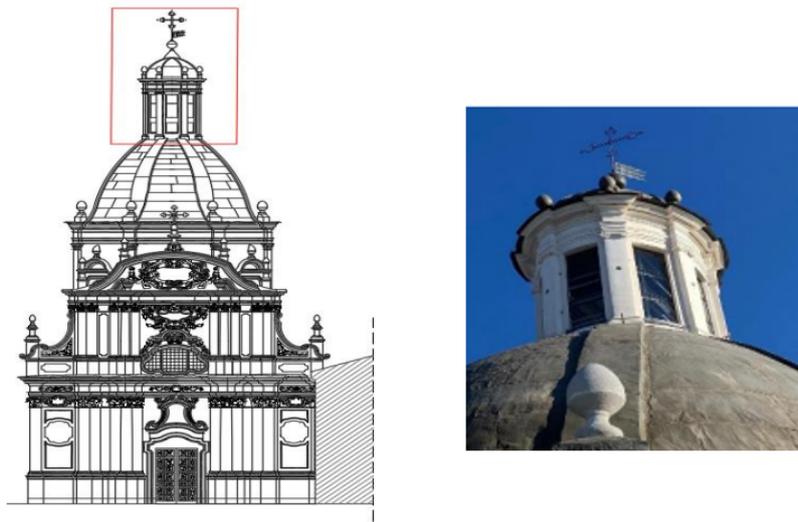


Figura 11: Prospetto principale della chiesa di Santa Caterina - Lanterino. [3]

3.1.4 Descrizione dello schema strutturale

Dal punto di vista strutturale, gli elementi verticali portanti sono costituiti principalmente da muri interni e perimetrali, insieme a dieci colonne principali che sostengono il tamburo, la cupola principale e la seconda cupola più piccola del presbiterio. Gli elementi orizzontali sono formati da volte e archi, anch'essi realizzati in muratura.

Lo schema strutturale principale che trasmette i carichi all'interno della chiesa prevede che le azioni siano trasferite alle fondamenta attraverso i quattro archi principali posizionati sui quattro lati della cupola ellittica. Questi archi, a loro volta, si appoggiano e scaricano il carico sui quattro pilastri principali. Anche per la seconda cupola, di dimensioni ridotte e di forma circolare, il carico viene trasferito al terreno attraverso volte e archi che successivamente lo trasmettono ai quattro pilastri presenti.

Il coro, invece, è collegato alla chiesa tramite aperture realizzate lungo la parete di divisione tra i due corpi. Questa parete divisororia è in muratura e ha uno spessore di circa

0,60 metri. La struttura portante verticale del coro è composta da dieci pilastri e da muri posizionati lungo i lati. Gli elementi orizzontali sono costituiti da un sistema di archi e volte. È importante notare che tutti gli archi, a causa di lavori di manutenzione precedenti, sono dotati di tiranti metallici. Lo schema strutturale trasferisce i carichi della grande volta e degli archi alle pareti di base e ai pilastri del coro.[3]

3.2 Storia del monitoraggio della chiesa

Prima del 2019, anno in cui sono iniziati i lavori di restauro, in particolare della facciata e del lanternino, le condizioni della struttura erano molto critiche, ovvero le infiltrazioni d'acqua avvenute nel sistema della cupola avevano non solo danneggiato gli affreschi interni, ma avevano anche causato, complice l'età della chiesa stessa, danni ingenti alla struttura, in particolare alla cupola, al lanternino e alla facciata, che presentava una frattura verticale a partire dal portone principale. L'acqua aveva inoltre comportato il distacco dell'intonaco dalla parete in molti punti e in alcuni casi addirittura efflorescenze saline.

In vista degli interventi di restauro, già dal 2010 era partita una campagna di studio finalizzata alla creazione di un iniziale modello FEM della chiesa, utile nell'individuazione delle componenti strutturali più influenti sul comportamento dinamico della struttura. Sempre nel settembre 2010, è stata condotta una campagna di test dinamici volti a individuare lo stato di salute della struttura, per individuare in tempo eventuali criticità della stessa, per un totale di tre diversi set-up, ognuno dei quali comprensivo di ben 18 canali di acquisizione.

Tale sistema di acquisizione ha permesso l'applicazione del processo informatico dell'SHM, al termine del quale sono stati individuati gli interventi necessari sulle parti più indebolite della struttura, la facciata e il lanternino appunto. Gli interventi che si sono svolti sono stati il rinforzo tramite delle piastre a L di acciaio del lanternino, in corrispondenza delle aperture delle finestre, andando a creare una sorta di scheletro metallico di sostegno dell'intera circonferenza del lanternino. Per quanto riguarda invece la facciata, è stato installato un telaio metallico collegato chimicamente al timpano all'interno, perché non fosse visibile dall'esterno, in modo da creare un comportamento a sistema collaborante, finalizzato a evitare il ribaltamento dell'intera parete muraria.

La campagna di monitoraggio di cui tratta il presente documento, eseguita tra dicembre 2022 e marzo 2023, è stata attuata con due obiettivi – in primis la ricerca di eventuali ulteriori parametri che possano influenzare il comportamento dinamico della struttura, e poi la ricerca di uno schema di sensori plausibile per la creazione di una rete di monitoraggio permanente sulla struttura, in modo che possa essere utilizzata come prevenzione per periodi

futuri.

3.2.1 Caso studio - i sensori

Definiti quindi come due obiettivi della campagna di studio del 2022/23 la ricerca di fattori che influenzino il comportamento della struttura e l'eventuale creazione di una rete di monitoraggio permanente sulla stessa, si può ora passare alla descrizione del set-up e degli strumenti che sono stati utilizzati.

Il set-up era composto da tre accelerometri posizionati sulla sommità del lanternino in modo da registrare le vibrazioni laterali e torsionali della struttura, due in direzione trasversale e una in direzione longitudinale – i sensori sono ad alta sensibilità sismica, come il modello ICP 393B12, 10 V/g, 0.15 to 1k Hz.



Figura 12: Una foto tipo dei sensori utilizzati (a sx) e (a dx) il posizionamento in pianta degli accelerometri con rispettiva direzione di registrazione.

Gli accelerometri sono collegati a un acquirente Dewesoft, che attua il conditioning, la sincronizzazione e la trasformazione da analogico in digitale dei dati, che poi vengono conservati in locale in un computer del laboratorio di Dinamica e Sismica presso il Politecnico di Torino.

Le registrazioni sono state avviate nel giorno 20 dicembre 2022 e sono terminate il 17 marzo 2023, ma è importante sottolineare come durante questa finestra temporale, molti periodi siano risultati vacanti dei dati, come riportato dalla figura successiva.

4 Prima fase del *Data Processing* - il *Preprocessing*

4.1 Introduzione al *Data Processing*

Indubbiamente una volta che il sistema di monitoraggio è stato messo in funzione, dopo l'accurata valutazione della struttura sotto analisi e del posizionamento dei sensori, al fine di arrivare all'identificazione dello stato di danno, è fondamentale la fase di trattamento dei dati, in cui la scelta del tipo di algoritmo da utilizzare è fondamentale.

I passaggi principali per la raccolta e analisi dei dati sono racchiusi nel modello Omnibus di Worden e Dulieu-Barton [8], di cui si riporta il flusso in figura:

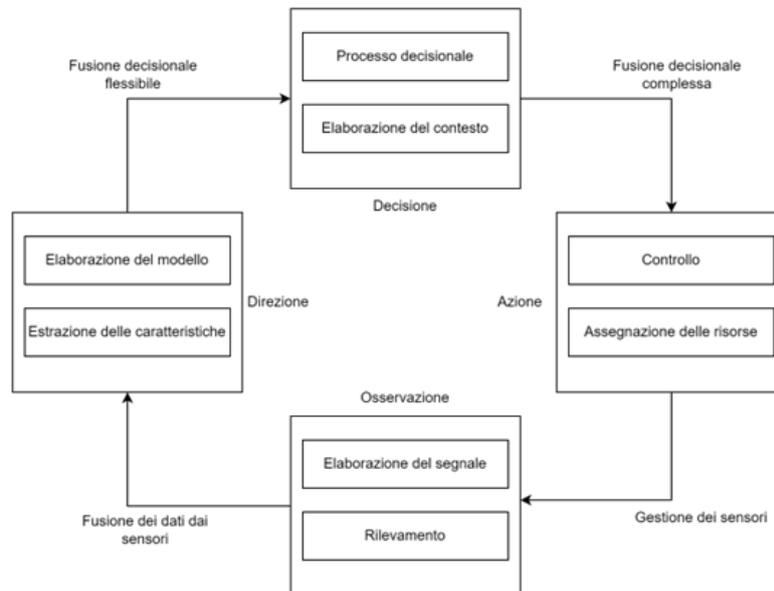


Figura 14: Il modello Omnibus per l'elaborazione dei dati.

In questo elaborato viene applicata l'identificazione dinamica, che può essere definita come l'insieme delle tecniche che permettono di individuare la risposta dinamica, nella for-

ma di frequenze proprie, coefficienti di smorzamento e modi di vibrare, della struttura in osservazione – universalmente è accettata la seguente classificazione:

- Per dominio;
- Sviluppo analitico;
- Numero di modi;
- Input e output;
- Tipologia di eccitazione.

Non è necessaria la descrizione di tutte queste tipologie in questo documento, ma ci si concentrerà sul metodo di classificazione nel dominio, poiché l'algoritmo applicato, lo *Stochastic Subspace Identification*, appartiene alla classe di dominio nel tempo.

Sebbene infatti le prime tecniche sviluppate appartenessero al dominio in frequenza, e nonostante la migliore applicabilità di questo tipo di tecnica a sistemi con un'alta densità di frequenze registrate, le tecniche di risoluzione nel dominio del tempo, nate in campi applicativi differenti, come la teoria dei sistemi, hanno cominciato a essere più utilizzate per risolvere i problemi legati al leakage, alla risoluzione in frequenza e alla densità modale elevata.

Esse si basano sul concetto che la risposta dinamica di una struttura possa essere considerata una serie temporale discreta, ovvero una rilevazione a intervalli temporali, da cui può essere poi costruito un sistema statistico. L'algoritmo qui applicato è del tipo *Stochastic Subspace Identification*, che può essere descritto come basato sulla manipolazione delle matrici contenenti i segnali mediante operazioni di algebra lineare. È di tipo *output-only*, in quanto le matrici di cui sopra possono essere determinate attraverso la sola conoscenza dei dati in uscita [12], associato alla costruzione del relativo filtro di Kalman.

I metodi che rientrano invece nelle analisi in tempo frequenza hanno l'obiettivo di descrivere contemporaneamente la densità di energia del segnale nel dominio appunto della frequenza e del tempo, trasformando segnali monodimensionali, dipendenti quindi da una sola di queste variabili, in bidimensionali, utilizzate principalmente per trattare segnali non stazionari.

Lo strumento analitico principale che si utilizza è la Short Time Fourier Transform, ovvero l'applicazione della trasformata di Fourier invece che all'intero segnale, a piccoli intervalli, ottenendo quindi molti spettri, ognuno relativo a un determinato intervallo temporale. Il lato negativo dell'applicazione di questa tecnica è l'insorgenza di molte imperfezioni nel procedimento in caso di segnali molto rumorosi o in presenza di molti punti di non stazionarietà – è necessario uno studio preliminare prima di applicare questo tipo di algoritmo.[10]

Di seguito verrà descritta l'applicazione di questi concetti al caso studio, con un ulteriore approfondimento nel procedimento utilizzato per il pre-trattamento dei dati.

4.2 Fase preparatoria dei dati

Il primo passaggio, propedeutico al trattamento effettivo dei dati provenienti dagli accelerometri, è stato la divisione dei file di misurazione, ottenuti da una misurazione in continuo (all'incirca uno al centesimo di secondo) in file più brevi, tutti della stessa lunghezza, corrispondente a una registrazione di venti minuti. Questo passaggio risulta fondamentale al fine di ottenere valori, in particolare sulla frequenza di oscillazione della struttura, confrontabili fra loro, in quanto considerati sul medesimo lasso temporale.

Si tratta della fase di caricamento dei dati originari salvati dalla strumentazione e di definizione della cartella di salvataggio dei file splittati alla fine del processo; viene inoltre anche definita la soglia del numero dei dati salvabili per ogni file, corrispondente a 120000, ovvero il numero di file contenuti in venti minuti, considerando il salvataggio continuo a monte come un file al centesimo di secondo.

```
% Caricamento dati e definizione delle variabili
threshold = 120000; % The number of values contained in 20 minutes.
filename = "dati registrati/Test.2023.02.28.115908";
savepath = 'Test.2023.02.28';
load(filename);
```

Applicazione della funzione *SPLIT VALUES*, commentata nel dettaglio nel paragrafo successivo, ovvero automatizzazione del processo di divisione dei file.

```
% Applicazione della funzione SPLIT VALUES

Data1_AI.2_splitted = split_values_by_threshold(Data1_AI.2, threshold);
Data1_AI.3_splitted = split_values_by_threshold(Data1_AI.3, threshold);
Data1_AI.4_splitted = split_values_by_threshold(Data1_AI.4, threshold);
Data1_time_AI.2_splitted = split_values_by_threshold(Data1_time_AI.2, threshold);
Data1_time_AI.3_splitted = split_values_by_threshold(Data1_time_AI.3, threshold);
Data1_time_AI.4_splitted = split_values_by_threshold(Data1_time_AI.4, threshold);
```

Questo ciclo *for* automatizza il salvataggio dei valori delle accelerazioni sui tre canali (corrispondenti ai due sensori e alle tre direzioni in cui vengono rilevati) insieme ai corrispondenti riferimenti temporali e al *sample rate*, ovvero la frequenza di campionamento,

che come riportato in precedenza corrispondeva a 100 valori al secondo. L'ultimo passaggio riguarda il salvataggio del nome del file, ottenuto con la successione numerica a valle della data di inizio della misurazione continua in soggetto.

```
% Salvataggio dati

for i = 1 : size(Data1_AI_2_splitted, 2)
    Data1_AI_2 = cell2mat(Data1_AI_2_splitted(i));
    Data1_AI_3 = cell2mat(Data1_AI_3_splitted(i));
    Data1_AI_4 = cell2mat(Data1_AI_4_splitted(i));
    Data1_time_AI_2 = cell2mat(Data1_time_AI_2_splitted(i));
    Data1_time_AI_3 = cell2mat(Data1_time_AI_3_splitted(i));
    Data1_time_AI_4 = cell2mat(Data1_time_AI_4_splitted(i));
    f = replace(filename, '.mat', '');
    save_file = strcat(savepath, '_', num2str(i), '.mat');
    save_file = fullfile(savepath, save_file);
    mkdir(savepath);
    save(save_file, 'Data1_AI_2', 'Data1_AI_3','Data1_AI_4',Data1_time_AI_2',
        'Data1_time_AI_3', 'Data1_time_AI_4','Sample_rate');
end
```

Di seguito è riportata la funzione che permette la divisione dei file di registrazione nei file singoli più piccoli e tutti della stessa dimensione. È necessario renderla una funzione di matlab, in quanto il processo di automatizzazione dell'acquisizione e apertura dei file è ottimizzato in questo modo, e non processando il codice all'interno dello stesso file per ogni iterazione. Si noti come vengono fornite alla funzione le due variabili *threshold*, la soglia numerica definita, ovvero 120000, e *arra*, ovvero il numero di valori presenti nell'intero file e quindi il numero di partenza per la valutazione del numero effettivo di file generati dalla singola registrazione – è un valore che differisce per ogni spezzona di registrazione, per cui

va indicato per ogni iterazione.

```
res = split_values_by_threshold(array, threshold) num_elements = numel(array);  
% Numero totale di elementi nell'array  
num_subarrays = ceil(num_elements / threshold); % Numero di sottoarray necessari  
res = cell(1, num_subarrays); % Cella per memorizzare i sottoarray  
%Ciclo for per la registrazione del numero di valori desiderati, a partire dalla fine  
del file precedente  
for i = 1:num_subarrays start_index = (i - 1) * threshold + 1;  
% Indice di partenza per il sottoarray  
end_index = min(start_index + threshold - 1, num_elements);  
% Indice di fine per il sottoarray  
resi = array(start_index:end_index); % Estrae il sottoarray dall'array originale  
end  
end
```

4.3 *Preprocessing*

Il *preprocessing* rappresenta la prima parte dell'analisi dati ed è necessario per procedere a ulteriori analisi, in modo da garantire la qualità, la coerenza e l'affidabilità dei dati, contribuendo così a sviluppare modelli accurati e a ottenere risultati affidabili.

Nel contesto del monitoraggio strutturale, il *preprocessing* assicura che le letture provenienti dai sensori distribuiti su una struttura siano accurate e disponibili per l'analisi. Questo processo coinvolge attività come la calibrazione dei sensori, la sincronizzazione temporale dei dati, la rimozione del rumore e l'adattamento delle letture per una migliore interpretazione. Una corretta fase di *preprocessing* è cruciale per garantire che i risultati dell'analisi successiva siano significativi e possano contribuire a una gestione più sicura ed efficace delle strutture sottoposte al monitoraggio.

Di seguito verrà riportata una descrizione sommaria dei passaggi principali che sono stati svolti dal codice utilizzato, in maniera da permettere la comprensione del processo logico e matematico di fondo di questa analisi, e vengono inoltre riportati alcuni risultati ottenuti nella parte applicativa.

4.3.1 Descrizione della procedura

La fase di seguito descritta ha come obiettivo l'attuazione una preelaborazione del segnale, con calcolo degli spettri di potenza e se richiesto anche una stima in tempo-frequenza.

Lo spettro di potenza è tipicamente la rappresentazione della quantità di energia distribuita per frequenza del segnale – solitamente infatti viene rappresentata da un grafico cartesiano, sul cui asse delle ascisse si riporta la frequenza del segnale e sull'asse delle ordinate l'ampiezza dello stesso, in modo da poter valutare la distribuzione spettrale complessiva e individuare le frequenze dominanti. Per calcolarlo, si applica una trasformata di Fourier al segnale, che lo scompone al fine di associare ad ogni frequenza la potenza corrispondente.

La stima in spettro-frequenza invece consente la rappresentazione spettrale del segnale, ma a differenza della trasformata di Fourier che ne fornisce una semplice raffigurazione, si aggiunge qui anche la variazione nel tempo, utile nel caso di variabili non stazionarie, con quindi un comportamento non lineare nel tempo.

Può poi essere eseguito, solo se necessario, il sottocampionamento del segnale, ovvero il procedimento di riduzione del numero dei campioni registrati, in modo da semplificare il segnale, rimuovendo ad esempio il rumore. È una tecnica da usare con la consapevolezza che se da un lato si alleggerisce così l'analisi, dall'altro può comportare perdita di informazioni chiave se non eseguita correttamente.

Da questo momento inizia il processing dei dati; in primis è necessario omogeneizzare i segni dei dati delle registrazioni con quelli del canale su cui sono state registrate, a cui segue la normalizzazione tramite la sottrazione del valor medio dei rispettivi canali.

Per quanto riguarda la compilazione dello spettro, è fondamentale calcolare i valori temporali per l'analisi, ovvero il passo temporale, il vettore dei tempi, il tempo finale dell'analisi del pacchetto di dati, insieme al passo delle frequenze e al vettore delle stesse.

Avviene poi l'applicazione di un filtro Butterworth, che ha lo scopo di andare a ripulire il segnale, cancellando o attenuando determinate frequenze del segnale, specificamente nel caso in analisi eliminando i valori al di sopra e al di sotto di una determinata soglia, stabilita a priori dal progettista. Il filtro presenta un andamento a fascia, con attenuazione più o meno accentuata a seconda del valore del filtro; nel caso dell'elaborazione in corso, si è stabilito che il 4 fosse adeguato, ciò comporta una maggiore selettività e una pendenza di attenuazione ripida, che possono essere fattori positivi, in quanto ripuliscono meglio il segnale, ma bisogna prestare attenzione con un valore del filtro così deciso perché può anche dall'altro lato portare a una complessità computazionale maggiore e una possibile perdita di dati significativi. Nel caso in esame ad esempio, le frequenze di taglio sono state impostate

a 0.5 in basso e 15 Hz in alto, valori scelti dopo un'analisi che comprendeva valori tra lo 0.01 e 30 Hz, sotto forma di varie combinazioni, quindi sia ampliando la banda del filtro, sia traslandola verso l'alto o verso il basso. Il filtro si basa sui parametri del vettore delle frequenze di taglio normalizzate, espresse dividendo le frequenze di taglio per metà della frequenza di campionamento. In ultimo, per ridurre gli effetti del ritardo di fase che è stato introdotto nel filtro viene eseguita la funzione *filtfilt* esegue un'ulteriore filtrazione in entrambe le direzioni, sia in avanti che indietro.

È stati calcolato poi lo spettro di potenza, cioè l'energia o la potenza del segnale alle diverse frequenze, tramite la trasformata di Fourier (DFT) discreta dei dati output, ottenuti in precedenza, che poi sono stati moltiplicati per il passo temporale, in modo da normalizzare l'ampiezza del segnale; infine la funzione *fftshift* trasla lo spettro in modo da ottenere al centro dello stesso le basse frequenze. La DFT ha la funzione di scomporre il segnale nel dominio delle frequenze, ottenendo quindi le sue componenti spettrali, e alla fine di questo passaggio viene assolutizzata la variabile risultante, ovvero l'ampiezza spettrale. Per ultimo, l'elevazione alla seconda potenza serve per ottenere lo spettro di potenza o l'energia spettrale del segnale.

Si passa quindi all'analisi in tempo frequenza del segnale, se ritenuta rilevante, partendo da un eventuale fattore di ingrandimento, utilizzato per modificare l'aspetto visivo del grafico, se ritenuto necessario per l'analisi. Devono essere definiti una serie di parametri necessari per l'analisi, tra cui una stima dei valori della frequenza, da cui si calcola poi la risoluzione in frequenza, e la lunghezza delle finestre usate nell'analisi, tramite l'applicazione del comando Hamming, oltre a definire il parametro *overlap*, che specifica la percentuale di sovrapposizione delle finestre – quanto più il valore sarà elevato, maggiore sarà tale sovrapposizione, che permette una rappresentazione più dettagliata delle caratteristiche del segnale che variano nel tempo. La funzione Hamming è predefinita in matlab come di tipo coseno sollevato, quindi con forma simmetrica con un picco centrale e decrescita graduale verso le code, assume valori tra 0 e 1. Viene usata per moltiplicare i segmenti del segnale durante l'analisi spettrale, in modo da ridurre gli effetti degli artefatti dati dall'utilizzo delle finestre rettangolari standard, e quindi rappresenta un buon compromesso tra la riduzione delle perdite spettrali causate appunto dalla finestra rettangolare, detti *leakage*, e la risoluzione in frequenza.

Nel caso del codice utilizzato in questo studio, è stato calcolato lo spettrogramma del segnale, ovvero la rappresentazione grafica della variazione della frequenza nel tempo e il processo di generazione si basa sul calcolo della trasformata di Fourier breve (STFT). Questa STFT consiste nell'applicazione della trasformata di Fourier a breve finestre temporali di codice, con successivo spostamento di tale finestra per ricoprire l'intera durata del segnale, mentre i risultati vengono ottenuti dalla combinazione di tutti gli output.

Si procede poi con la normalizzazione del risultato, dividendo ciascun elemento dello spettrogramma ampliato per il massimo valore assoluto presente nello stesso – la funzione di questo passaggio è di rendere tutti i coefficienti spettrali compresi nell’intervallo $[0;1]$, in relazione alla massima ampiezza presente. Per ultimo si attua la somma cumulativa delle normalizzazioni dei coefficienti spettrali per tutti i canali.

4.4 *Preprocessing* nel caso studio

Nel processo di trattamento dei dati, partendo dai valori registrati nei tre mesi attivi (Dicembre 2022 – Marzo 2023), sono stati attuati una serie di step fondamentali per la preparazione dei file alla fase centrale del documento, ovvero il *post processing*. I file delle registrazioni sono stati quindi prima separati in spezzoni corrispondenti a 20 minuti di elaborazione – con conseguente eliminazione di alcune parti in eccedenza – in modo da essere confrontabili tra loro, e poi passati per una fase di *preprocessing* – una sorta di pulitura dei file da rumore e *outliers* evidenti. Infine devono attraversare una fase di identificazione, che può comprendere la sincronizzazione dei dati temporali, l’identificazione di pattern significativi, il rilevamento di anomalie e la generazione di rapporti dettagliati, ad esempio.

Il codice utilizzato per l’analisi dei dati provenienti dai sensori sulla chiesa di Santa Caterina, dopo il caricamento dei dati, filtra prima i dati ottenuti attraverso un filtro passa-banda, avendo trovato i valori limite per le frequenze accettabili, poi passa al calcolo dello spettro di potenza con finestra di Welch e con il metodo della trasformata di Fourier.

Il primo grafico ottenuto dal codice è stato quello che rappresenta il calcolo della densità di potenza con il metodo di Welch, che suddivide il segnale in segmenti sovrapposti, calcola la PSD (*Power Spectrum Density* – densità di potenza spettrale) per ciascun segmento e quindi ne calcola la media. Questo approccio migliora la stabilità e la precisione della stima della PSD rispetto a una stima diretta utilizzando l’intero segnale. Il grafico raccoglie le frequenze sull’asse delle ascisse e le PSD calcolate tramite questo metodo sull’asse delle ordinate.

Di seguito sono riportati i due grafici che si ottengono dall’analisi in tempo frequenza, il primo relativo ad un singolo sensore, nel caso in esame il canale 1Y, il secondo relativo alla media. Anche qui, sull’asse delle ascisse si vede come venga segnato il tempo, sulle ordinate la frequenza.

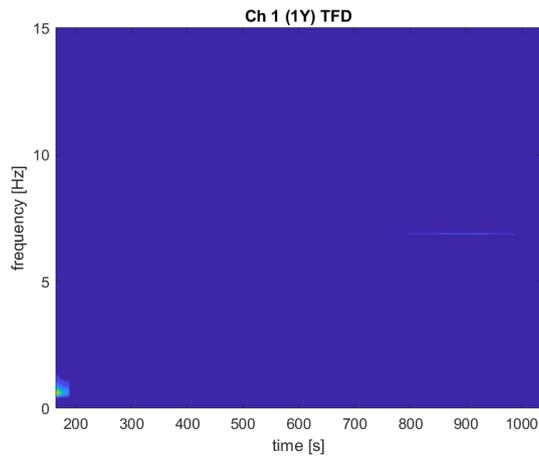


Figura 15: Il grafico della TFD per il primo canale.

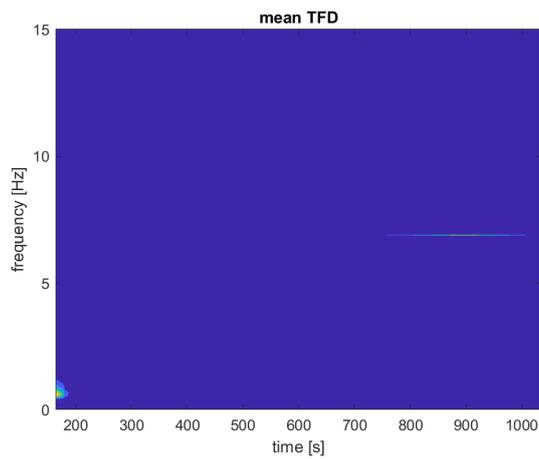


Figura 16: Il grafico della TFD per la media dei canali.

Di seguito viene riportata una parte del risultato ottenuto, come tabella, dal workspace – in particolare nella prima colonna si ritrovano i riferimenti temporali del momento in cui sono stati registrati i dati, nelle altre tre colonne i valori delle accelerazioni, la prima rappresenta la direzione trasversale del primo sensore, la seconda quella longitudinale, sempre del primo sensore, e la terza la direzione trasversale del secondo sensore.

5 Seconda fase del *Data Processing* - l'identificazione dinamica

L' *identification* dei dati nel contesto del monitoraggio strutturale rappresenta il cuore del trattamento dei dati. Si tratta di un processo che essenzialmente consiste nella raccolta, l'analisi e l'interpretazione dei dati provenienti dai sensori distribuiti sulla struttura in esame, che può essere nella forma di un ponte, un edificio o una diga. L'identificazione dinamica viene utilizzata nei casi, come quello del presente documento, in cui non si conoscano a priori le caratteristiche del sistema o nei casi in cui si vogliano analizzare quelle stesse, se note, per ricercare eventuali cambiamenti causati da variazioni o evoluzioni dello stato di danno.

Tali sensori registrano una serie di parametri, come vibrazioni, deformazioni, spostamenti o accelerazioni ad esempio, al fine di valutare lo stato di salute e l'integrità della struttura. Il *processing* dati comprende attività come la sincronizzazione dei dati temporali, l'identificazione di pattern significativi, il rilevamento di anomalie e la generazione di rapporti dettagliati. Questo processo fornisce quindi dati importanti al fine di attuare manutenzione preventiva e analisi sulla sicurezza strutturale, contribuendo così a garantire anche la sicurezza delle persone e la protezione del patrimonio architettonico.

5.1 *Stochastic Subspace Identification (SSI-CVA)*

Nel presente capitolo verrà descritto in maniera teorica l'algoritmo di identificazione utilizzato per il processamento dei dati preparati per la chiesa di Santa Caterina, il *Stochastic Subspace Identification* (SSI-CVA) – la caratteristica degli algoritmi di tipo *subspace* è di essere un approccio utilizzato per stimare un modello di sistema lineare e invariante nel tempo a partire da dati che includono dati di *output-only* del sistema. Questo metodo sfrutta la proiezione ortogonale di una matrice che contiene i dati più recenti nello spazio determinato dalle righe di una matrice contenente dati molto meno recenti.[13]

5.2 Descrizione del procedimento

Verrà però descritto tematicamente in seguito il procedimento computazionale seguito.

5.2.1 Gestione delle variabili

Dopo la necessaria dichiarazione di variabili, la prima parte del codice che è stato utilizzato per l' *identification* è dedicata alla gestione dei parametri dei dati e all'estratto di segmenti degli stessi, rilevanti per analisi successive. Si inizia con un fattore di decimazione, utile a ridurre il numero di dati, in modo da ottenere una nuvola trattabile in maniera più semplice, poi si calcola la frequenza di campionamento, ovvero la quantità di campioni raccolti al secondo.

In input sono stati forniti i dati raccolti, che sono di tipo accelerazione, dal momento che i sensori applicati sulla struttura erano accelerometri, trovandoci nel caso di una *Dynamic Structural Monitoring*; la dimensione di tale input è data da un lato dal numero di campioni raccolti, mentre nell'altra direzione dal numero di canali disponibili, ovvero il numero di strumenti effettivamente funzionanti applicati sulla struttura, che vengono anche denominati canali. Si rende inoltre necessaria la preparazione e la segmentazione dei dati, in modo da poter permettere l'elaborazione degli stessi in parti.

Il codice passa poi alla definizione di diversi parametri con la finalità di andare a definire lo spazio stato del sistema in analisi, ovvero la descrizione di un sistema dinamico tramite l'attribuzione appunto di alcune variabili che creano uno spazio vettoriale in cui è definito tale sistema. Il modello quindi identificato rispetto ai valori definiti si utilizza con scopo di ottimizzazione, controllo e previsione per quanto riguarda il comportamento di sistemi dinamici complessi.

5.2.2 Diagramma di stabilizzazione

Inoltre sono anche necessari da definirsi i parametri di un diagramma di stabilizzazione, dal momento che i metodi di identificazione dinamica soffrono un problema alquanto fondamentale, ovvero la dichiarazione dell'ordine del sistema o dei gradi di libertà – trovandosi nel caso di strutture esistenti, dagli infiniti gradi di libertà, si può comprendere come sia una decisione cruciale per l'analisi.

La procedura quindi prevede la ripetizione dell'identificazione per ordini diversi del sistema, che causa però l'inconveniente della presenza di modi spuri, o *noise modes*, che vanno poi isolati per evitare errori; ciò viene attuato con il confronto di alcuni parametri per ogni identificazione, tra cui si trovano: lo smorzamento, ovvero la capacità di dissipare l'energia vibratoria durante l'oscillazione, espresso in percentuale nel codice; lo scarto in frequenza, cioè la differenza minima di frequenza tra modi successivi rilevanti; il coefficiente di accoppiamento modale, che rappresenta la correlazione tra due modi di vibrare differenti, un alto valore è indice di forte correlazione, uno basso il contrario; l'indice di occorrenza, cioè una soglia minima di occorrenza di una determinata modalità di vibrazione affinché essa possa essere considerata come valida, ciò funziona come una sorta di filtro per i modi di vibrare più importanti o significativi.

È interessante citare come la stabilizzazione sia considerata su più variabili, che venga quindi applicata separatamente a più valori che singolarmente valutano una condizione specifica:

- Stabilità per frequenza;
- Stabilità per smorzamento;
- Stabilità secondo il coefficiente di accoppiamento modale;
- Stabilità complessiva, che riporta alla stabilità soddisfatta per tutti i modi di cui sopra.[10]

5.2.3 *Cluster analysis*

L'ultima parte tratta la *cluster analysis*, una tecnica di analisi dei dati che si basa sul raggruppamento di elementi omogenei, stabiliti appunto tramite i dati in input, che vanno selezionati attentamente in modo da ottenere dei raggruppamenti utilizzabili per le analisi successive. Sono quindi importanti da definire i metodi di collegamento dei valori all'interno di un cluster, oltre agli estremi dei dati da suddividere, tutti valori di progetto che devono essere inizializzati nel codice.

In particolare si tratta l'applicazione dell'algoritmo di *clustering* FCM (Fuzzy C-Means), una tecnica di clustering dei dati elaborata negli anni '70 che si basa sul concetto che i singoli dati possano appartenere a più di un cluster, non prevedendo quindi una separazione netta dei cluster in base alle similitudini, ma considerando che i dati come variabili che possono possedere caratteristiche varie che sanciscono quindi l'appartenenza del punto a più di un gruppo.[14]

5.3 *L'Identification* nel caso studio

5.3.1 Dichiarazione delle variabili

Il primo passaggio nel caso studio è sicuramente la definizione dei parametri e l'analisi delle variabili presenti nel codice.

Di seguito è prima riportata la legenda per le sigle attribuite alle variabili, poi la tabella riassuntiva di tutte le variabili di input.

FIR	Finite impulse response
AmplFFT	Fast Fourier transform
STFT	Short time Fourier transform

Figura 17: Le sigle delle variabili

Nome variabile	Identificazione
Data1_AI_1_FIR_Filter	NO DATA
Data1_time_AI_1_FIR_Filter	NO DATA
2 -> 1Y	
Data1_AI_2	ACC(1Y)
Data1_time_AI_2	TEMP(1Y)
Data1_AI_2_FIR_Filter	ACC(1Y) filtrata con Finite impulse response
Data1_time_AI_2_FIR_Filter	TEMP(1Y) relativo a Finite impulse response
Data1_AI_2_FIR_Filter_AmplFFT	
Data1_time_AI_2_FIR_Filter_AmplFFT	
Data1_X_AI_2_FIR_Filter_AmplFFT	
Data1_AI_2_FIR_Filter_STFT	
Data1_time_AI_2_FIR_Filter_STFT	
Data1_X_AI_2_FIR_Filter_STFT	
3 -> 1X	
Data1_AI_3	ACC(1X)
Data1_time_AI_3	TEMP(1X)
Data1_AI_3_FIR_Filter	ACC(1X) filtrata con Finite impulse response
Data1_time_AI_3_FIR_Filter	TEMP(1X) relativo a Finite impulse response
Data1_AI_3_FIR_Filter_AmplFFT	
Data1_time_AI_3_FIR_Filter_AmplFFT	
Data1_X_AI_3_FIR_Filter_AmplFFT	
Data1_AI_3_FIR_Filter_STFT	
Data1_time_AI_3_FIR_Filter_STFT	
Data1_X_AI_3_FIR_Filter_STFT	
4 -> 2Y	
Data1_AI_4	ACC(2Y)
Data1_time_AI_4	TEMP(2Y)
Data1_AI_4_FIR_Filter	ACC(2Y) filtrata con Finite impulse response
Data1_time_AI_4_FIR_Filter	TEMP(2Y) relativo a Finite impulse response
Data1_AI_4_FIR_Filter_AmplFFT	
Data1_time_AI_4_FIR_Filter_AmplFFT	
Data1_X_AI_4_FIR_Filter_AmplFFT	
Data1_AI_4_FIR_Filter_STFT	
Data1_time_AI_4_FIR_Filter_STFT	
Data1_X_AI_4_FIR_Filter_STFT	

Figura 18: Riassunto degli input

5.3.2 Descrizione dell'analisi

Il codice quindi, dopo le necessarie dichiarazioni delle variabili, parte dall'identificazione di modelli a sistemi di diversi ordini, che rappresentano i gradi di libertà del sistema, ovvero si tratta della stima dei parametri che permettano la creazione di un modello matematico atto a stabilire il comportamento di un sistema dinamico, utilizzando i dati sperimentali in entrata e uscita.

Vengono quindi applicati i criteri di stabilizzazione ai risultati ottenuti dal passaggio precedente, che portano alla riduzione dei modi instabili, tramite il confronto con i coefficienti di smorzamento ammessi, sia massimi che minimi; poi, una volta ripuliti i modi stabili, si vanno ad analizzare i delta per la frequenza, smorzamento e coefficiente di accoppiamento per stabilire se vengono così soddisfatti i criteri appunto di stabilizzazione – ciò permette di valutare la coerenza e l'affidabilità dei modi identificati.

Dopo aver valutato la stabilità generale dei modi, per cancellare i modi spuri, si passa a una valutazione più nel dettaglio per valori specifici, che vanno a setacciare i risultati, rendendoli più affidabili per i passaggi ancora da svolgere.

Nel caso del codice che è stato applicato, i risultati sono stati suddivisi in categorie in base ai parametri che sono stati utilizzati; in primis, viene effettuata la classificazione, e una conta, dei modi in stabili, instabili e spuri, che risultano da cancellare, tramite l'analisi della matrice dei risultati; vengono poi suddivisi i modi instabili in base al criterio che non è rispettato, quindi nel caso in esame se la frequenza, lo smorzamento o il coefficiente di accoppiamento modale; i modi stabili invece vengono raccolti in un'unica matrice, che contiene non solo i dati relativi al modo di vibrare ma anche quelli relativi ai criteri di stabilizzazione che hanno seguito.

Nella parte di codice successiva, vi è l'applicazione dell'algoritmo di *clustering* FCM (Fuzzy C-Means), in cui vengono subito trattati i dati, in modo da selezionare e poi preparare i dati idonei per essere trattati con questo algoritmo, tramite un'organizzazione per frequenza crescente, l'aggiunta di una colonna dei pesi da assegnare ai dati prima del clustering e il calcolo delle dissimilarità del MAC, attuato con l'ausilio della creazione di una matrice.

Per procedere con questo algoritmo, viene ricercata la dissimilarità totale tra modi, in modo da poter considerare quali valori non saranno simili per nessuna parte e poter procedere con la ricerca di parziali similarità quindi tra i valori trovati; il procedimento quindi

prevede l'aumento della differenza accettabile sia del MAC che della frequenza, oltre al calcolo appunto della dissimilarità totale; viene anche definita la distanza minima accettabile, ovvero il delta massimo tra valori basato sia sul peso assegnato ai singoli dati sia sulle differenze accettabili calcolate prima.

Si procede poi con l' *Agglomerative hierarchical cluster tree*, un dendrogramma gerarchico dei cluster, ovvero un algoritmo per l'identificazione dei cluster che si basa sulla creazione di cluster diminuendo a ogni ciclo la soglia di similarità, quindi che vanno restringendosi di passaggio in passaggio. La scelta progettuale rientra poi nella soglia di taglio, ovvero il valore di similarità considerato corretto, per cui si andrà a tracciare una retta parallela all'asse delle ascisse in corrispondenza di tale valore, che definisce come validi i cluster contenuti al di sotto di essa.

Un esempio di dendrogramma ricavato è riportato in figura, che fa riferimento a uno slot temporale di registrazione pari a 10 minuti, ovvero metà del valore del file di dati in input.

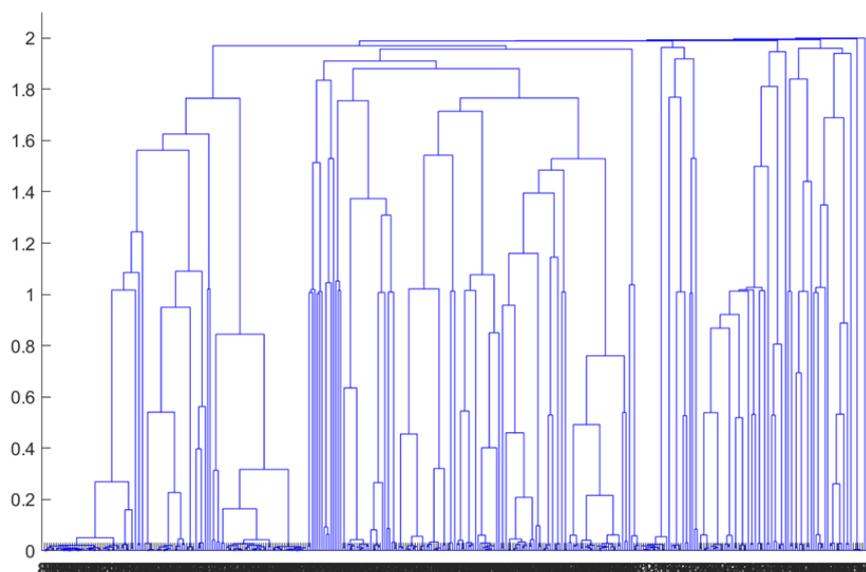


Figura 19: Dendrogramma con slot temporale di 10 minuti.

Nel codice è stato poi previsto un paragrafo che vada a smistare i valori dei cluster ottenuti e la selezione delle varie frequenze, con la registrazione anche di quelle scartate, in

particolare selezionando i valori in base alle forme modali predominanti.

Il codice passa poi al calcolo delle statistiche del cluster totale e dei singoli cluster, ad esempio il numero di occorrenze dello stesso valore all'interno del cluster o la media dei valori delle frequenze o del MAC, oltre a ottenere il numero totale di cluster. È importante sottolineare come poi venga applicato un ulteriore filtro, che scarta tutti i cluster che presentano un numero di frequenze al loro interno minore di una determinata soglia, registrandone però poi le statistiche, in modo da avere informazioni riguardo anche ai cluster che vengono eliminati.

Per ultimo il codice esegue alcune operazioni post-elaborazione e genera i grafici e le immagini di risultato. Avviene per prima l'identificazione degli *outliers*, ovvero i valori che statisticamente risultano essere anomali, ovvero distanti da tutte le altre osservazioni di cui si dispone, e al contrario dei cluster selezionati come candidati, cioè accettati e utilizzabili per l'identificazione finale; procede poi con la rimozione dei suddetti *outliers*, per poi definire gli estremi dell'ordine del sistema.

In conclusione, si tratta quindi di un esempio di un processo di identificazione strutturale che mira a determinare le caratteristiche dinamiche di un sistema strutturale utilizzando dati di ingresso/uscita. Questo processo fornisce una panoramica dettagliata delle caratteristiche dinamiche del sistema sotto esame, aiutando a identificare e comprendere i modelli strutturali coinvolti.

In figura sono infatti riportati gli ulteriori grafici di risultato, il primo che mette in relazione i valori dello smorzamento e le frequenze, individuando tutti i modi risultati come candidati e i valori in corrispondenza dei modi considerati stabili.

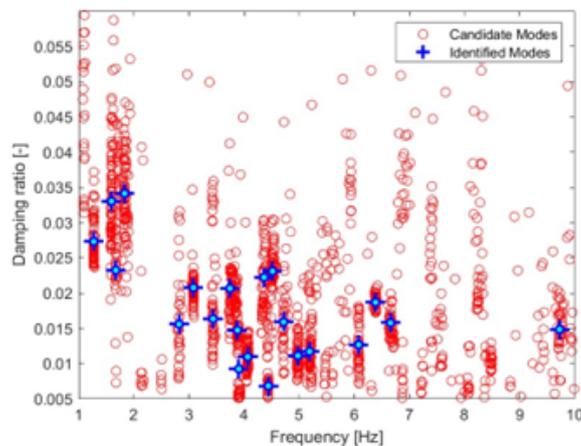


Figura 20: Grafico smorzamento-frequenza, con i modi candidati e quelli identificati.

Il secondo grafico invece riporta gli ordini del sistema per tutti i modi di vibrare, sia quelli considerati spuri che quelli validati e considerati quindi stabili, secondo il concetto che l'identificazione andava ripetuta per ordini diversi, procedimento che comportava l'identificazione non solo dei modi puri, ma anche di molti *noise modes*, che qui sono riassunti in un unico diagramma.

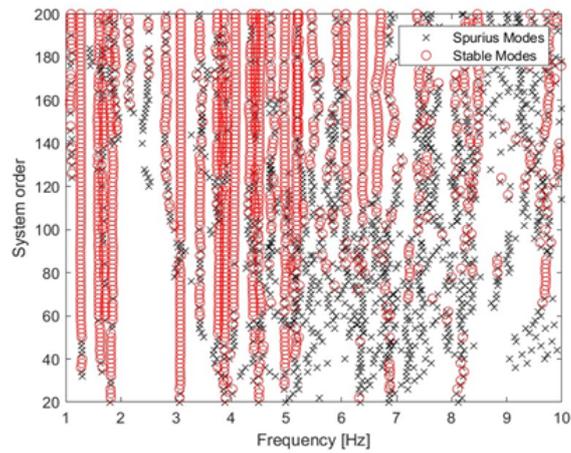


Figura 21: Grafico ordine del sistema-frequenza, con i modi spuri e stabili.

6 Terza fase del *Data Processing - Post-processing* con classificazione manuale

Si approda così all'ultimo passaggio del processo di trattamento dei dati, il *post-processing*, una fase fondamentale perché, sebbene i passaggi precedenti fossero quelli volti all'effettiva analisi dei dati provenienti direttamente dai sensori, rappresenta il momento in cui i risultati vengono elaborati in modo da essere resi interpretabili e utilizzabili dal tecnico. Essa precede la comunicazione dei risultati e la presa di decisioni, in modo che questi possano quindi diventare utili per la progettazione, la costruzione, la gestione e la manutenzione delle infrastrutture e delle opere civili.

Il seguente capitolo si incentra sulla descrizione del processo logico e pratico dietro alla parte di classificazione manuale dei dati ottenuti dall'identificazione per lo svolgimento della post elaborazione. L'idea concettuale su cui si basa questo procedimento è la ricerca di una logica nel processo, che possa essere individuata e quindi automatizzata tramite la creazione di un codice matlab apposito – la ricerca quindi dell'aggregazione dei cluster, in modo da isolare le singole classi di dati e individuare così i modi propri di vibrare della struttura, a cui sono associati i dati relativi alle frequenze, smorzamenti e coefficiente MAC.

È intuitivo come non fosse necessario attuare tale procedimento su tutti i file ottenuti dal passaggio precedente, ma che fosse sufficiente ricavare un campione di file su cui applicarlo – nello specifico è stato ritenuto valido attuare la classificazione manuale su un campione di 450 file circa, ottenuti ricercando i picchi positivi e negativi di temperatura lungo il periodo di registrazione, rappresentanti alla fine poco meno del 20% del totale.

Di seguito è riportato il ragionamento dietro la ricerca di questo campione di dati, e successivamente è descritto il procedimento di classificazione, completo delle varie fasi logiche, anche erronee, che caratterizzano ogni studio di tipo sperimentale.

6.1 Ricerca dei picchi di temperatura

In virtù della necessità di attuare una parte della codificazione per cluster in maniera manuale, in modo da poter individuare il processo logico e matematico per automatizzare il processo tramite linguaggio macchina e di conseguenza ottimizzarlo e velocizzarlo, andando non solo a ridurre i tempi di elaborazione ma anche eventuali errori umani che sopraggiungono nella ripetizione dello stesso procedimento in maniera meccanizzata, si è posto il quesito sulla decisione di quali file sarebbero stati presi come campione per il processo di codificazione manuale, per poi confrontarli con gli stessi risultati ottenuti dal codice matlab.

Naturalmente la scelta era ristretta a una rosa di eventi caratterizzanti, che quindi consideravano per lo più eventi atmosferici e naturali che avrebbero potuto creare movimenti interessanti nella struttura, o che si suppone abbiano correlazione con il movimento della stessa, e quindi si è deciso di continuarne l'analisi.

La decisione è ricaduta su un tipo di evento "naturale" ciclico e ripetibile, ovvero le variazioni di temperatura, quindi i picchi relativi della stessa, sia verso il lato dei massimi che dei minimi.

1. Primo metodo

Il primo approccio, rivelatosi fallimentare al termine del passaggio però, è stato quello di ricercare, tramite i dati della temperatura registrati in gradi Celsius ogni mezz'ora forniti in un file .mat, compatibile quindi con il codice matlab, i picchi di temperatura massimi e minimi assoluti sullo spazio temporale in cui erano state effettuate le registrazioni.

Si è reso poi necessaria la pulizia di tali dati ottenuti, poiché le registrazioni dei sensori risultano essere, come già descritto in precedenza, lacunose in alcuni spazi temporali, per cui era necessario trovare la correlazione tra i dati effettivamente registrati e i picchi di temperatura, inutili altrimenti se privi del corrispettivo valore registrato in quell'istante dal sensore.

Ottenuti quindi i dati di temperatura e accelerazioni corrispondenti, si è notato come, naturalmente, basandosi su registrazioni avvenute con uno step temporale di mezz'ora la stessa giornata si ripresentava come picco in positivo o negativo più volte, rendendo quindi poco rappresentativo il campione di dati analizzabile manualmente, perché spesso sovrapposto ad altri picchi e di conseguenza poco vario.

2. Secondo metodo

Dovendo quindi ricercare di creare un campione di dati più rappresentativo, si è dovuto

procedere in maniera differente, andando a modificare la logica della ricerca dati. La differenza quindi rientrava nella ricerca di picchi di temperatura, ma questa volta non globali, ma locali.

Per rendere più semplice il processo di ricerca, si è optato per la divisione in spezzoni temporali da circa 15 giorni (con l'eccezione di dicembre, perché la raccolta dei dati dai sensori è iniziata nella sua terza decade), e quindi con l'obiettivo di ricavare due picchi massimi e due minimi per ogni mese solare, in modo da avere una distribuzione più omogenea dei file da analizzare in maniera manuale.

Di seguito si riporta uno spezzone di codice che è stato scritto per attuare questo procedimento:

```
load("dati.mat"); %caricamento del file delle temperature
Date=DatetimeInt

%Divisione dei file caricato nei mesi solari (ricerca manuale)
Dicembre = [Date(1:548)]
DicTemp= [VariablesInt(1:548,1)]
Gennaio = [Date(549:964)]
GennTemp= [VariablesInt(549:964,1)]
Febbraio = [Date(3017:3380)]
FebbTemp= [VariablesInt(3017:3380,1)]
Marzo = [Date(3381:4195)]
MarTemp= [VariablesInt(3381:4195,1)]

% Definizione il numero di valori massimi/minimi da estrarre
numMassimi = 50;
numMinimi = 50;

% Estrazione dei valori massimi e minimi
[valoriMassimi, indiciMassimi] = maxk(MarTemp, numMassimi);
% Estrapola i valori massimi e i loro indici
[valoriMinimi, indiciMinimi] = mink(MarTemp, numMinimi);
% Estrapola i valori minimi e i loro indici

%Stampa dei valori risultati
```

I dati ottenuti sono stati raccolti quindi in un file excel, di cui in seguito è riportata la

tabella riassuntiva, in cui si può quindi notare come si è cercato di avere una distribuzione, il quanto più possibile omogenea, mantenendo l'idea di picchi di temperatura il più possibile lontani dalla media della temperatura del periodo.

Le colonne contengono il riferimento al mese in esame, la data e l'ora relativi alla temperatura del picco, riportata anch'essa, seguono la cartella di riferimento in cui trovare il file* relativo al picco e i numeri dei file da analizzare manualmente, considerando all'incirca quindici file precedenti e successivi, per arrivare a un totale di analisi di circa 450 file complessivi.

Mese	Data	Orario	Temperatura [°C]	Cartella di riferimento	Numero di file
dic-22	27-dic	h 13.30	12.8	26-dic	dal 47 al 73
	23-dic	h.14.00	8.6	22-dic	dal 51 al 75
gen-23	05-gen	h 18.30	11.00	03-gen	dal 155 al 182
	03-gen	h 22.30	8.6	03-gen	dal 23 al 52
feb-23	21-feb	h 15.00	14.2	21-feb	dall'1 al 28
	25-feb	h 13.00	14.2	24-feb	dal 72 al 101
mar-23	11-mar	h 15.30	22.10	09-mar	dal 110 al 139
	05-mar	h 13.30	15.90	28-feb	dal 351 al 380

Tabella – Individuazione dei file da analizzare manualmente, riportanti data, ora e temperatura e cartelle di riferimento - Picchi massimi di temperatura

Mese	Data	Orario	Temperatura [°C]	Cartella di riferimento	Numero di file
dic-22	31-dic	h 7.00	1.5	29-dic	dal 96 al 125
	21-dic	h 7.30	2.7	20-dic	dal 43 al 72
gen-23	06-gen	h 2.30	2.9	03-gen	dal 183 al 209
	09-gen	h 8.00	5.3	03-gen	dal 411 al 438
feb-23	26-feb	h 13.30	2.1	24-feb	dal 146 al 175
	28-feb	h 21.00	2.6	28-feb	dal 13 al 60
mar-23	01-mar	h 2.00	2.5	28-feb	come sopra
	03-mar	h 5.00	3.5	28-feb	dal 124 al 153

Tabella – Individuazione dei file da analizzare manualmente, riportanti data, ora e temperatura e cartelle di riferimento - Picchi minimi di temperatura

*NOTA: la cartella di file viene riportata perché si ricorda come le registrazioni iniziali dei sensori siano state raccolte in file dal nome della data di inizio della registrazione, ma hanno lunghezze e numeri di file al loro interno differenti, anche su un arco temporale di più giorni. I primi passaggi propedeutici al processing dei dati erano proprio

stati lo splitting dei dati raccolti in spezzoni da 20 minuti l'uno, in modo che fossero confrontabili tra loro e per superare la diversità temporale dei file registrati originali.

Di seguito è riportata anche la tabella cronologica su cui sono stati segnalati i file soggetti a controllo manuale, in base ai picchi di temperatura riportati sopra e poi individuati con i numeri in tabella. La funzione di tale immagine è rendere visivamente più chiaro il fatto di aver cercato di distribuire quanto più possibile il campione di dati di riferimento – rispettivamente in blu e in rosso i picchi in basso e in alto di temperatura, con l'ora di inizio e fine registrazione. Si può notare come alcune volte i campioni scelti non si estendessero per esattamente per 5 ore, corrispondenti a 15 spezzoni di registrazione, prima e dopo il momento individuato come picco, ma le motivazioni si basano sulla mancanza di un numero tale di dati o sulla sovrapposizione con un altro punto di misurazione campionato, data la brevità, da questo punto di vista, del periodo di registrazione totale dei dati.

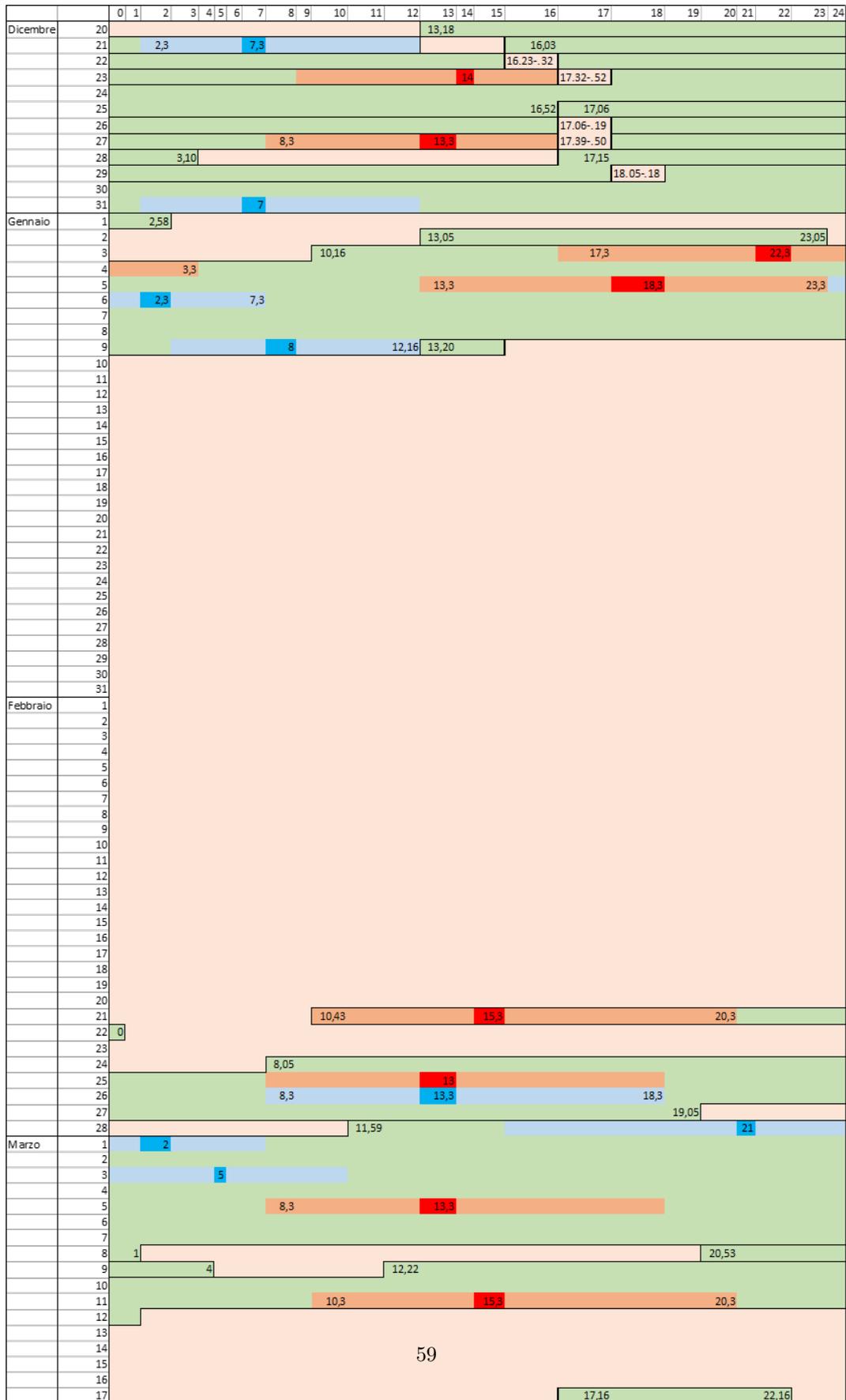


Figura 22: Individuazione dei picchi di temperatura e relativi file da analizzare nel crono-

6.2 Concetti fondamentali

Impostando l'ambiente di lavoro, nel caso in analisi, è importante già sottolineare come la fase precedente di identificazione dei dati aveva dato come risultato, per ogni singolo file di input, due file separati, sequenziali temporalmente, che nella fase di post processing vanno considerati come coppia – ogni analisi dovrà essere considerata nulla se applicata a uno solo dei file in oggetto, nonostante tale file possa essere sintatticamente corretto. È quindi estremamente importante impostare un parametro di controllo iniziale, per scartare, se dovessero essercene, cartelle di file in cui appunto non vi sia la presenza di entrambi gli spezzoni di identificazione.

Stabilito quindi che non è richiesto effettuare l'analisi per un'identificazione singola, nel caso in analisi, ma sarebbe possibile modificando un solo parametro di controllo posizionato appositamente per rendere più semplice la scelta dell'opzione da seguire, si passa alla parte di *post processing*, che comincia con una definizione di alcuni parametri utili alla creazione di grafici – in primis i grafici di stabilizzazione e i *clustering*.

6.2.1 La stabilizzazione

Dal momento che l'ordine del sistema in esame durante il procedimento di data analysis è l'incognita fondamentale, non conoscendone a priori il valore, e poiché dalla sua determinazione dipendono molti fattori determinanti, come ad esempio il numero di modi di vibrare identificati per la struttura, è importante che, per valutare i risultati ottenuti per quanto riguarda le caratteristiche dinamiche del sistema, questi siano indipendenti dall'ordine del sistema stesso.

Proprio in virtù di questo principio si attua un controllo di stabilità dei risultati stessi, appunto nella forma di un diagramma di stabilizzazione, in cui si controlla la stabilità dei modi del sistema per individuare quali siano solamente numerici e quali siano i modi puri.

Ciò è attuato tramite i tre parametri fondamentali di frequenza, forma modale e smorzamento, di cui si vanno a valutare i delta del singolo valore (ordine n) con quello di ordine successivo ($n+1$) che devono rientrare nel valore stabilito da progetto secondo una tolleranza.

6.2.2 Il clustering

Dopo aver individuato i modi stabili del sistema, è necessario andare a raggruppare i modi in cluster al fine di ottenere i modi rappresentativi – ciò è stato attuato tramite il linkage clustering, che nella fase manuale si è basato solo sulla differenza, in percentuale per alcune variabili e in valore assoluto per altre, a seconda della scelta progettuale, tra valori vicini, mentre nella parte di ricerca automatica si basata sulla ‘distanza’ tridimensionale tra valori, per ottenere le massime similitudini, ovvero vicinanze nello spazio tridimensionale di analisi.

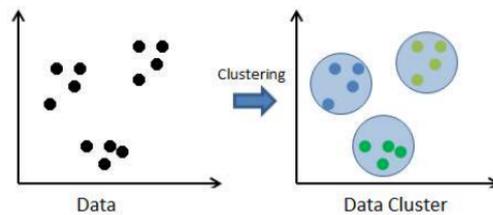


Figura 23: Esempio di cluster analisi in 2D.

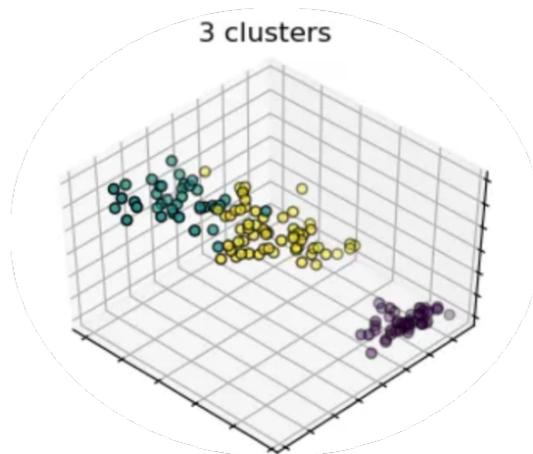


Figura 24: Esempio di cluster analisi in 3D.

6.2.3 Mediazione

I passaggi successivi riguardano l'ordinamento e la pulizia dei dati nelle matrici di risultati; infatti ad esempio i valori ottenuti nella matrice degli autovettori delle forme modali devono

essere normalizzati, dividendo ogni colonna per il massimo valore assoluto presente nella colonna – ciò garantisce che i dati siano nella stessa scala – e poi si attua un cambio di segno, moltiplicazione per -1, in modo da assicurarsi che i dati delle modalità di forma strutturale siano coerenti in termini di fase.

La fase di mediazione automatica tratta appunto la fase di ricerca dei valori medi, e le rispettive deviazioni standard, per le frequenze, le ammortizzazioni e i modi di vibrare dei cluster individuati, in modo da ottenere un singolo valore di riferimento per ogni singolo modo di vibrare proprio della struttura – nel caso che il cluster fosse composto da un solo elemento non sarebbe necessario il calcolo della deviazione standard, ma nel caso in analisi è stato deciso di scartare i cluster singoli per evitare di cadere in errori di valutazione dati da modi che potrebbero avere alta probabilità di essere spuri, per cui non si rientra mai in tale categoria.

6.2.4 I risultati

Ottenuti tutti i risultati ordinati, è possibile quindi ricavare i grafici di interesse, come i grafici di dispersione, ottenuti con l'ordine del sistema sull'asse delle y e sia le modalità sull'asse delle x, sia quelle ridotte, andando a sovrapporre sullo stesso grafico i valori delle due variabili, indicate ovviamente con simboli diversi, in modo da poter confrontare i risultati. Il risultato finale risulterà essere un grafico su piano cartesiano in cui saranno sovrapposti sia i modi "candidati", quindi tutti i modi individuati come possibilmente validi, sia quelli effettivamente identificati come tali.

Di seguito vengono riportati i grafici corrispondenti.

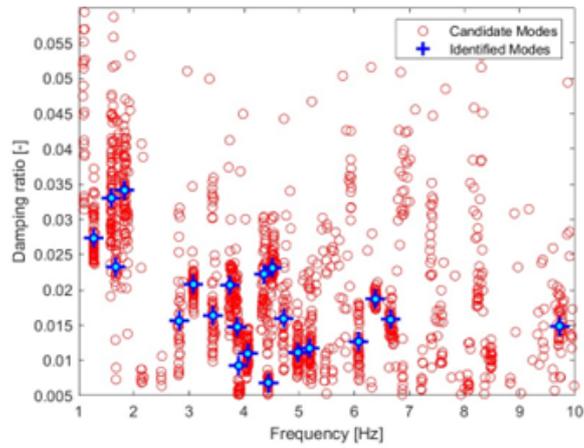


Figura 25: Grafico di stabilizzazione, con i modi candidati e quelli identificati.

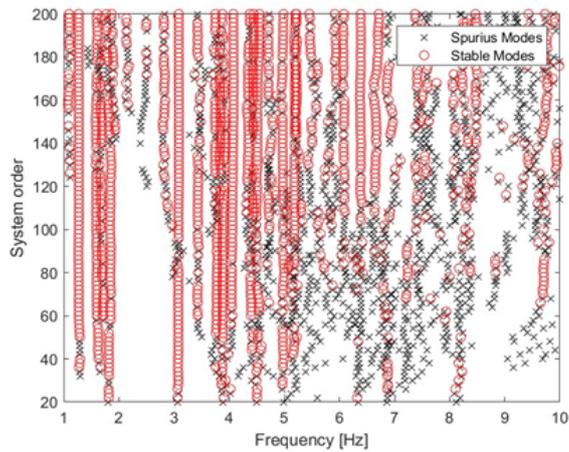


Figura 26: Grafico ordine del sistema-frequenza, con i modi spuri e stabili.

6.3 Classificazione manuale basata sulle frequenze e sul MAC

Dopo aver individuato i file da classificare, la fase manuale del post processing inizia dopo la prima parte del codice, ovvero quella di aggregazione e combinazione dei file ottenuti dalle identificazioni – due per ogni spezzona di dati registrati.

La classificazione manuale è necessaria in quanto il numero di classi finali non è noto a priori, e si ricaverà per additività di un'unità per ogni ciclo del codice, che viene considerato per

ogni cluster individuato. Per stabilità dei risultati, i cluster contenenti un solo valore alla fine del processo di analisi non vengono considerati validi al fine della classificazione.

Importante specificare che nel seguente capitolo sono riportati tutti i passaggi che sono stati effettuati basandosi su intervalli e delta anche erronei e che sono stati affinati nel corso del processo – per completezza espositiva si riportano anche dei valori reali come esempio della differenza fra i vari passaggi. È stato deciso che sarebbe potuto essere un approccio completo alla descrizione del processo empirico, poiché si è appunto basato sull’osservazione dei risultati ottenuti in maniera diversa, per arrivare al termine del processo con i valori considerati più verosimili.

L’intero processo manuale è anche servito come base per la creazione di un codice per l’automatizzazione, quindi anche le diverse soglie utilizzate sono state fonte di riflessione per la stesura del processo logico in linguaggio macchina.

Il procedimento logico, considerato in questo caso come un filtro a setaccio, quindi graduale, passaggio per passaggio, è riportato di figura – vengono riportati nel paragrafo che segue i cambiamenti del flusso logico che sono stati presi in considerazione con l’osservazione del processo empirico.

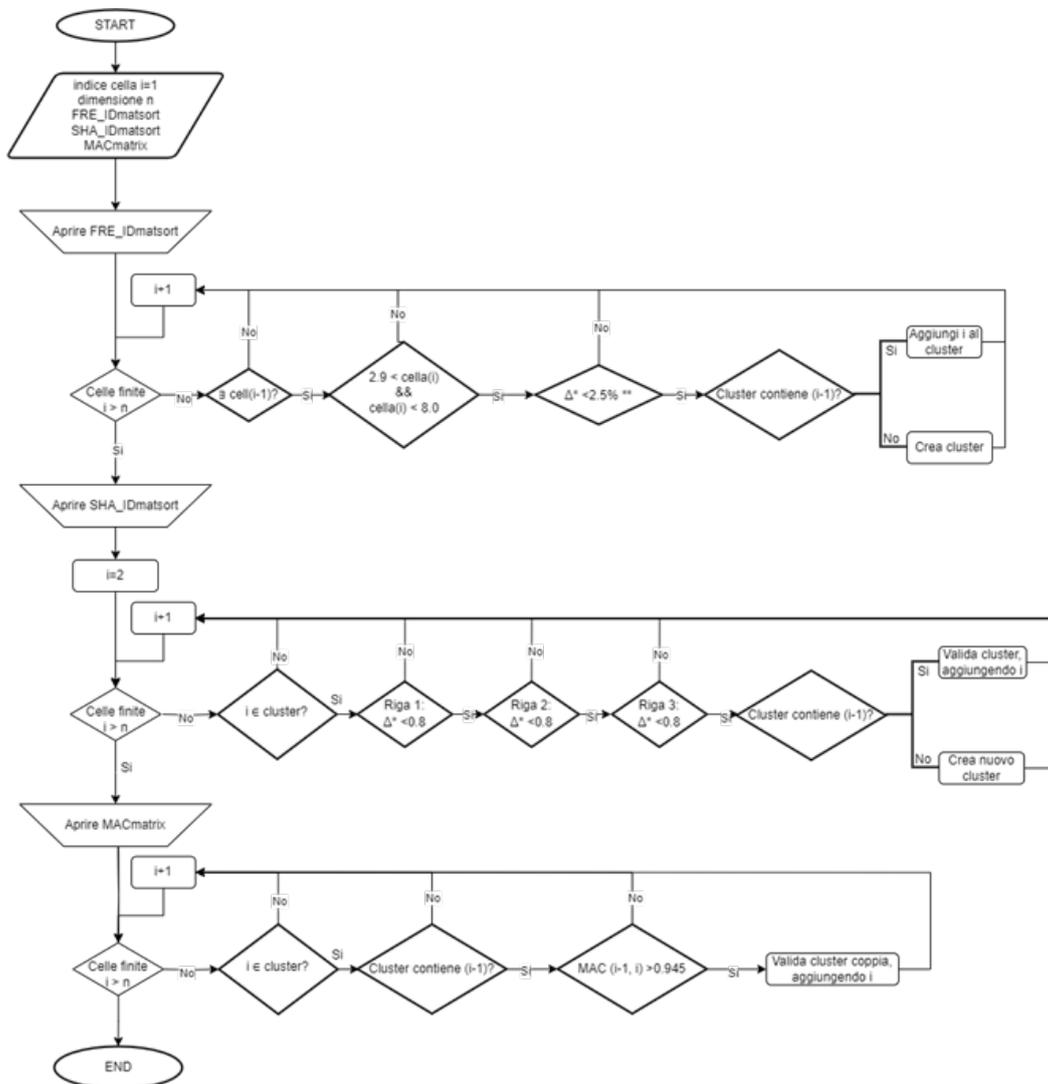


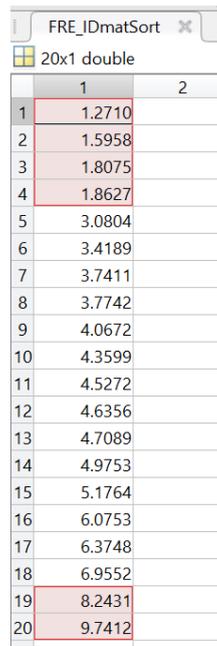
Figura 27: Flow chart della logica dell'individuazione manuale dei cluster

Il primo passaggio, al raggiungimento del break point preimpostato, è l'analisi della variabile "FRE_IDmatSort", ovvero l'array che raccoglie i valori delle frequenze registrate e ripulite da rumore e altri disturbi, ma ordinate in maniera crescente.

Inizialmente i valori assoluti delle frequenze al di sopra del valore di 8 Hz sono stati scartati, in quanto non attribuibili correttamente alla struttura, possibilmente proprie del ponteggio o frutto di errori sistematici. In una fase temporale successiva, dopo che erano stati analizzati diversi file di registrazione, è stato deciso che i valori di frequenza anche molto bassi, al di sotto di 2.9 Hz, non erano da considerarsi, in quanto, se anche propri della

struttura, potevano causare una propagazione dell'errore nella fase del codice immediatamente successiva, nello specifico nel calcolo della media dei valori delle frequenze nei singoli cluster al fine di ottenere un solo valore rappresentativo della stessa.

Nella pratica il vettore ripulito risulta come segue:



	1	2
1	1.2710	
2	1.5958	
3	1.8075	
4	1.8627	
5	3.0804	
6	3.4189	
7	3.7411	
8	3.7742	
9	4.0672	
10	4.3599	
11	4.5272	
12	4.6356	
13	4.7089	
14	4.9753	
15	5.1764	
16	6.0753	
17	6.3748	
18	6.9552	
19	8.2431	
20	9.7412	

Figura 28: Vettore “FRE_IDmatSort” con in evidenza i valori da scartare

Nel passaggio successivo, inizia l'individuazione dei cluster – il primo approccio è stato quello di considerare un delta assoluto, il cui valore si è andato a definire con lo studio di molti file sequenziali, nel range tra 0.4-0.8 Hz, per poi assestarsi sul lato più basso attorno allo 0.4 Hz :

	1	2
5	3.0804	
6	3.4189	
7	3.7411	
8	3.7742	
9	4.0672	
10	4.3599	
11	4.5272	
12	4.6356	
13	4.7089	
14	4.9753	
15	5.1764	
16	6.0753	
17	6.3748	
18	6.9552	

Figura 29: Cluster validati con $\Delta = 0.4$ Hz

Il secondo approccio, più corretto invece, si è basato sul concetto che l'incidenza di un delta pari a 0.4 Hz sui valori all'inizio del vettore era molto più importante che sui valori al termine del vettore, per cui è stata adottata la differenza di tipo percentuale, a partire da un valore del 6%, scendendo al 4% (dove già si riscontravano valori plausibili) per arrivare al 2.5%, che matematicamente rappresentano intervalli molto più piccoli di delta tra i valori.

	1	2
5	3.0804	
6	3.4189	
7	3.7411	
8	3.7742	
9	4.0672	
10	4.3599	
11	4.5272	
12	4.6356	
13	4.7089	
14	4.9753	
15	5.1764	
16	6.0753	
17	6.3748	
18	6.9552	

Figura 30: Cluster validati con $\Delta = 4\%$ rispetto al valore precedente

FRE_IDmatSort		
20x1 double		
	1	2
5	3.0804	
6	3.4189	
7	3.7411	
8	3.7742	
9	4.0672	
10	4.3599	
11	4.5272	
12	4.6356	
13	4.7089	
14	4.9753	
15	5.1764	
16	6.0753	
17	6.3748	
18	6.9552	

Figura 31: Cluster validati con $\Delta = 2.5\%$ rispetto al valore precedente

Ricavati in questo modo i cluster corretti secondo il primo passaggio della frequenza, che venivano registrati a mano attraverso gli indici delle celle corrispondenti, si è attuato un processo simile ma confrontando ora tra i soli valori accettati in cluster dal primo passaggio i corrispondenti valori della matrice “SHA_IDmatSort”, ovvero la matrice di tre righe che raccoglie i valori degli autovettori dei modi di vibrare.

In questo caso la condizione di accettazione dei cluster era più complessa e richiedeva che tutti e tre i valori delle tre righe avessero uno scarto non superiore a una determinata cifra, posta prima attorno allo 0.5, per arrivare gradualmente fino allo 0.8. Il motivo per cui la scelta è ricaduta su un valore assoluto, e non in percentuale, era la maggiore flessibilità che tali valori potevano assumere per essere considerati appartenenti allo stesso cluster. Di seguito sono riportati due esempi per sottolineare la differenza tra i valori scelti come soglie, all’inizio dell’elaborazione e il valore effettivamente scelto come valido.

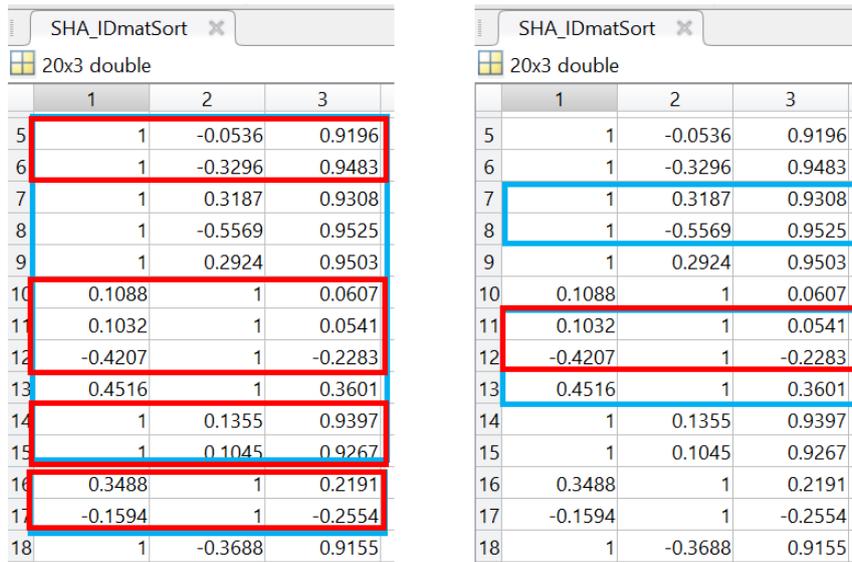


Figura 32: Cluster validati (in rosso) con $\delta = 0.6$ rispetto al valore precedente, a sinistra con i cluster delle frequenze ricavati dal delta in valore assoluto, a destra con il delta in percentuale (in blu)

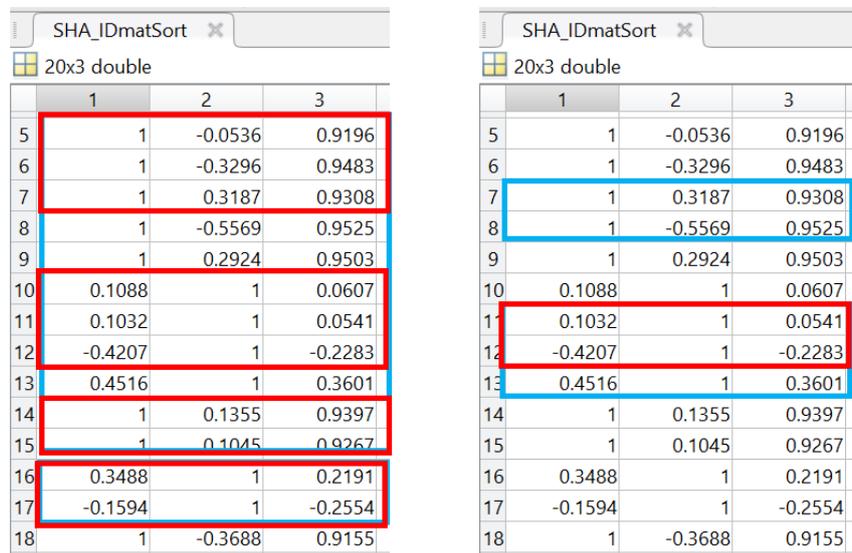


Figura 33: Cluster validati (in rosso) con $\delta = 0.8$ rispetto al valore precedente, a sinistra con i cluster delle frequenze ricavati dal delta in valore assoluto, a destra con il delta in percentuale (in blu)

Tra i controlli obbligatori da svolgere in questo passaggio, è di particolare importanza l'attenzione al segno matematico dei valori raccolti nella matrice. Il procedimento successivo di elaborazione dei dati, infatti, richiede il calcolo della media dei valori di frequenza, autovettori e del coefficiente MAC per ogni cluster individuato, in modo da ottenere un solo valore caratteristico per ogni variabile che rappresenti il modo di vibrare – in questo procedimento è intuitivo come anche il segno attribuito alle variabili rappresenti un valore importante e proprio per un errore di tipo strumentale a volte il segno riportato nella matrice è di valore contrario rispetto a quello reale. L'individuazione di tale errore è dunque fondamentale per ottenere risultati corretti nel processo di elaborazione dei dati.

Per evitare la propagazione dell'errore strumentale è sufficiente confrontare i valori, per ogni riga, considerandone la media, a coppie – qualora si notassero valori discostanti dalla media intuitivamente corretta, come ad esempio valori vicini allo zero assoluto o valori vicini all'unità, su tutti e tre i canali di registrazione (le tre righe) si riterrà necessario moltiplicare la colonna corrispondente per il valore negativo dell'unità, in modo da cambiare segno.

In particolare per la struttura oggetto di questo documento, la Chiesa di Santa Caterina, tale procedimento non è stato necessario, in quanto tale errore strumentale non era presente in nessuna occasione.

L'ultimo passaggio necessario consisteva nel controllo del valore del coefficiente MAC, un filtraggio però molto più ferreo dei precedenti, che infatti in ultime fasi di analisi del procedimento si è dibattuto se renderlo primo passaggio. Teoricamente, seguendo la logica della validazione a setaccio, i risultati non sarebbero cambiati per nulla, anche modificando i passaggi di ricerca dei cluster, in una sorta di analogia con la proprietà commutativa.

Il controllo del MAC è stato il passaggio più lineare e immediato, con accettazione dei valori con un coefficiente pari o maggiore a 0.95 – unica modifica è stata l'accettazione di valori fino allo 0.945, che per arrotondamento sarebbero comunque arrivati alla soglia prestabilita, in modo da non perdere valori che potenzialmente risultavano corretti dai passaggi precedenti. Per attuare questo controllo, è necessario nella matrice dei MAC controllare la cella con numero di riga pari all'indice del primo valore del cluster da confrontare e con numero di colonna pari al secondo valore del cluster, e ripetere per ogni combinazione di indici. È interessante sottolineare come tale matrice risulta essere simmetrica per proprietà, e che sulla sua diagonale principale i valori sono pari all'unità, poiché rappresentano la somiglianza del singolo valore a se stesso (quindi il 100%), mentre la sovradiagonale e la sottodiagonale contengono i valori interessati da questo controllo, che appunto sono uguali, in virtù della simmetria della matrice.

MAC_MATRIX		6	7	8	9	10	11	12	13	14	15	16	17	18
5		0.9621	0.9295	0.8861	0.9393	0.0066	0.0053	0.2060	0.2157	0.9810	0.9866	0.1141	0.0995	0.9497
6		0.7997	1	0.9781	0.8157	0.0131	0.0151	0.3787	0.0802	0.8930	0.9061	0.0219	0.2442	0.9988
7		0.7997	1	0.6694	0.9995	0.1173	0.1118	0.0409	0.4657	0.9830	0.9769	0.3299	0.0029	0.7744
8		0.9781	0.6694	1	0.6884	0.0676	0.0720	0.5240	0.0191	0.7853	0.8029	1.3361e-07	0.3807	0.9859
9		0.8157	0.9995	0.6884	1	0.1043	0.0991	0.0488	0.4449	0.9878	0.9824	0.3102	0.0055	0.7909
10		0.0131	0.1173	0.0676	0.1043	1	0.9999	0.7084	0.8470	0.0470	0.0383	0.9304	0.8445	0.0208
11		0.0151	0.1118	0.0720	0.0991	0.9999	1	0.7157	0.8408	0.0435	0.0351	0.9261	0.8507	0.0234
12		0.3787	0.0409	0.5240	0.0488	0.7084	0.7157	1	0.3231	0.1069	0.1212	0.4488	0.9448	0.4109
13		0.0802	0.4657	0.0191	0.4449	0.8470	0.8408	0.3231	1	0.3378	0.3176	0.9800	0.4806	0.0646
14		0.8930	0.9830	0.7853	0.9878	0.0470	0.0435	0.1069	0.3378	1	0.9995	0.2142	0.0336	0.8731
15		0.9061	0.9769	0.8029	0.9824	0.0383	0.0351	0.1212	0.3176	0.9995	1	0.1970	0.0417	0.8874
16		0.0219	0.3299	1.3361e-07	0.3102	0.9304	0.9261	0.4488	0.9800	0.2142	0.1970	1	0.6188	0.0141
17		0.2442	0.0029	0.3807	0.0055	0.8445	0.8507	0.9448	0.4806	0.0336	0.0417	0.6188	1	0.2697
18		0.9988	0.7744	0.9859	0.7909	0.0208	0.0234	0.4109	0.0646	0.8731	0.8874	0.0141	0.2697	1

Figura 34: Cluster validati con $MAC_i=0.945$, ottenuti dal delta di 0.8 per i cluster individuati con delta in valore assoluto della frequenza

MAC_MATRIX		6	7	8	9	10	11	12	13	14	15	16	17	18
5		0.9621	0.9295	0.8861	0.9393	0.0066	0.0053	0.2060	0.2157	0.9810	0.9866	0.1141	0.0995	0.9497
6		0.7997	1	0.9781	0.8157	0.0131	0.0151	0.3787	0.0802	0.8930	0.9061	0.0219	0.2442	0.9988
7		0.7997	1	0.6694	0.9995	0.1173	0.1118	0.0409	0.4657	0.9830	0.9769	0.3299	0.0029	0.7744
8		0.9781	0.6694	1	0.6884	0.0676	0.0720	0.5240	0.0191	0.7853	0.8029	1.3361e-07	0.3807	0.9859
9		0.8157	0.9995	0.6884	1	0.1043	0.0991	0.0488	0.4449	0.9878	0.9824	0.3102	0.0055	0.7909
10		0.0131	0.1173	0.0676	0.1043	1	0.9999	0.7084	0.8470	0.0470	0.0383	0.9304	0.8445	0.0208
11		0.0151	0.1118	0.0720	0.0991	0.9999	1	0.7157	0.8408	0.0435	0.0351	0.9261	0.8507	0.0234
12		0.3787	0.0409	0.5240	0.0488	0.7084	0.7157	1	0.3231	0.1069	0.1212	0.4488	0.9448	0.4109
13		0.0802	0.4657	0.0191	0.4449	0.8470	0.8408	0.3231	1	0.3378	0.3176	0.9800	0.4806	0.0646
14		0.8930	0.9830	0.7853	0.9878	0.0470	0.0435	0.1069	0.3378	1	0.9995	0.2142	0.0336	0.8731
15		0.9061	0.9769	0.8029	0.9824	0.0383	0.0351	0.1212	0.3176	0.9995	1	0.1970	0.0417	0.8874
16		0.0219	0.3299	1.3361e-07	0.3102	0.9304	0.9261	0.4488	0.9800	0.2142	0.1970	1	0.6188	0.0141
17		0.2442	0.0029	0.3807	0.0055	0.8445	0.8507	0.9448	0.4806	0.0336	0.0417	0.6188	1	0.2697
18		0.9988	0.7744	0.9859	0.7909	0.0208	0.0234	0.4109	0.0646	0.8731	0.8874	0.0141	0.2697	1

Figura 35: Cluster validati con $MAC_i=0.945$, ottenuti dal delta di 0.8 per i cluster individuati con delta in percentuale

Si può facilmente notare come l'approccio con il delta in percentuale per il calcolo delle frequenze sia di gran lunga più conservativo e cautelativo nei confronti dell'aggregazione dei diversi modi di vibrare. Allo stesso tempo, questo tipo di scelta progettuale può causare la perdita di informazioni rilevanti, come si osserva in maniera esemplare nel caso proposto, in quanto mentre utilizzando il primo approccio si ricavano ben tre cluster validi, nel secondo caso, le soglie troppo rigide impediscono la riuscita del processo di filtraggio, dando come risultato il *null*. Si può quindi intuire che il processo logico per la riuscita dell'analisi dell'intero pacchetto di file sia stato frutto di una serie di *trial-and-error**, che hanno dato

come risultati finali quelli che contenessero intuitivamente il minor numero di errori o di incongruenze. Non si può garantire in questi casi di ottenere risultati perfettamente corretti, ci si deve accontentare dello scorporo graduale di errori sistematici o di progetto, fino ad ottenere risultati compiacenti e plausibilmente lo specchio più vicino alla realtà possibile.

Dopo aver individuato quindi i cluster corretti, quindi i valori che possono essere raccolti e mediati in modo da ottenere i risultati per ogni modo proprio di vibrare della struttura, vengono inseriti nel codice automatico gli indici relativi ai singoli cluster, insieme a un conto additivo per ottenere al termine del ciclo il numero totale di classi, come riportato dal codice qui sotto:

```
%Manual classification of modes based on frequency and MAC
% This cell "class_mode" must be filled manually since the number of class
% are unknown at priori. Class with just 1 mode are neglected for safety
mm = 1;
class_modemm = [1:2];

mm = mm+1;
class_modemm = [3:4]; [..]
```

Il codice procede quindi con la valutazione della media delle variabili per ogni singolo cluster individuato e salva i valori nelle cartelle corrispondenti.

Il procedimento di tipo manuale è stato ripetuto per tutti i 450 file individuati dalla ricerca dei picchi di temperatura, e al termine del processo i risultati sono stati raccolti in un unico file riportante la data e l'ora del momento centrale del singolo file, la temperatura corrispondente e i valori sia della media che della deviazione standard delle frequenze, degli autovettori e del coefficiente MAC.

*Il procedimento empirico di *trial-and-error* consiste nella ricerca del risultato corretto tramite cicli di applicazione, con valutazione dei risultati ottenuti e del passaggio successivo da applicare attuata a ogni iterazione.

7 Automatizzazione del procedimento di *post-processing*

Il nucleo di questo documento è rappresentato dall'elaborazione più complesso di questo procedimento, ovvero la resa in automatico con linguaggio macchina del procedimento effettuato manualmente nel capitolo precedente. Come può essere intuitivo, è stato necessario ricercare dei percorsi logici che permettessero di trasformare il flusso manuale in automatico, comportando quindi alcune variazioni importanti del flow chart, di cui è riportato uno schema in figura.

Il modo più veloce per l'elaborazione del processo è stato ritrovato nella scrittura di più funzioni informatiche per ottenere in ogni passaggio una singola validazione, di cui quindi in seguito sono riportate le parti salienti, con riferimenti al codice stesso.

7.1 Commento al codice

La prima parte del codice contiene alcune informazioni generali riguardo al processo che si andrà ad eseguire, come l'ordine, tramite appunto la definizione delle funzioni, in cui saranno eseguite le varie operazioni fondamentali.

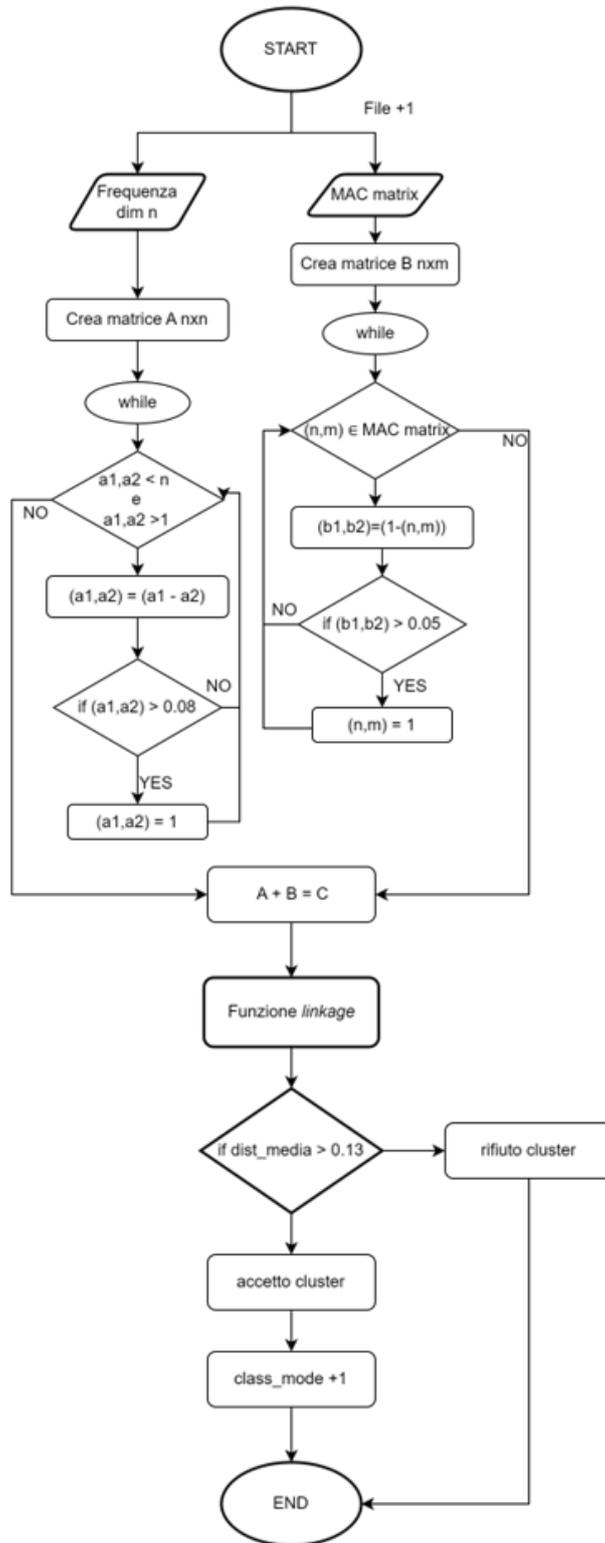


Figura 36: Diagramma di flusso del processo automatizzato

7.1.1 *Post-processing*

Inizialmente, come scelta progettuale per tenere conto del procedimento automatico del codice durante il suo svolgimento, viene definito un *execution registry*, un file che contenga solamente il nome dei singoli file di registrazione di cui sia stato completato il passaggio di *post-processing*, come un registro utile nel caso di blocchi del codice in attività o per qualche errore di computazione. In caso infatti di necessità di fermare il procedimento e poi di ricominciare, i file già processati non saranno duplicati perché il codice, leggendo il loro nome sul registro, non li considererà – è notevole il risparmio temporale.

```
registry_file = "execution_registry.txt";  
% Registry file check  
if exist(registry_file, 'file') == 0  
fileID = fopen(registry_file, 'w');  
fclose(fileID);  
disp('Execution registry file created.');
```



```
registry = fileread(registry_file);
```

Dopo aver definito le cartelle di input e output dei dati da elaborare, è stato necessario inserire un piccolo controllo per individuare i dati corretti da elaborare; per fare ciò è stato comunicato al programma che i file corretti da considerare possedevano un nome dato da alcune parole seguite da un numero – la data di registrazione, definita nei passaggi precedenti – e quindi risulta quindi chiaro al codice che tutti altri tipi di file che possono essere contenuti nella cartella non sono da elaborare con questo iter.

Allo stesso modo è importante sottolineare come sia stato impostato un ciclo for per prevedere il processamento di tutti i file contenuti nelle cartelle di input, e che provvede anche a calcolare la lunghezza del file .mat di input dati, dato fondamentale per il filtraggio successivo, che avviene all'interno delle funzioni, spiegate in seguito.

```

parentFolder=C:/Users/Desktop/identification lab/tesi/Output;
files = dir(parentFolder);
savePath = C:/Users/Desktop/TESI/POST PROCESSING/ExportData
/Results;
groups = group(files);
mkdir(savePath);

% Iterate over each item in the parent folder
keys = groups.keys;
for i = 1:length(keys)
    key = keys(i);
    key = key1;
    fileGroup = groups(key);
    pattern = '/d4_/d2_/d2)_N(/d+)';
    match = regexp(key, pattern, 'tokens', 'once');

    date = match1;
    iter = str2num(match2);
    saveFolder = fullfile(savePath, strcat('data_filt_', date, '_', num2str(iter)));
    for j = 1:length(fileGroup)
        registry = fileread(registry_file);
        fileName = fileGroup(1).name;
        if contains(registry, fileName)
            disp(['File', fileName, "already processed"]);
            break;
        else
            mkdir(saveFolder)

```

La riga di codice contenuta nel paragrafo successivo è il richiamo alla funzione principale di *postprocessing*, ovvero lo stesso codice, a parte il richiamo aggiunto per l'automatizzazione descritto nel paragrafo successivo, del procedimento manuale – cioè l'analisi dei dati vera e propria.

```

% Process the file here
postprocessing_fn_auto(parentFolder, saveFolder, key, length(fileGroup));
fileID = fopen(registry_file, 'a');
fprintf(fileID, '%s/n', fileName);
fclose(fileID);
end
end
end

```

La funzione descritta successivamente è semplicemente un meccanismo di ordinamento dei dati in input in base al loro nome, ovvero la data di registrazione.

```

function groups = group(files)
% Crea una mappa (map) per raggruppare le stringhe per data
map = containers.Map();

% Loop attraverso l'array di stringhe
for i = 1:length(files)
% Estrai la data dalla stringa usando regexp
file = files(i);
match = regexp(file.name, pattern, 'tokens', 'once');

% Se la data è stata estratta correttamente
if isempty(match)
data = match1;
% Controlla se esiste già una chiave per questa data nella mappa
if isKey(map, data)
% Aggiungi la stringa all'array esistente
map(data) = [map(data), files(i)];
else
% Crea una nuova chiave nella mappa e inizializza l'array
map(data) = [files(i)];
end
end
end
end

```

```

% Visualizza i gruppi di stringhe per data
keys = map.keys;
for i = 1:length(keys)
data = keys(i);
strings = map(data1);
end

groups = map;
end

```

7.1.2 *Post-processing_fn*

Si passa ora alla parte di elaborazione dei dati concreta, il procedimento già descritto nel capitolo relativo al post processing – questa parte ancora non contiene automatizzazioni, ma solo un richiamo alla funzione di individuazione delle classi, il processo che nel capitolo precedente era stato effettuato manualmente.

La variabile debug ha la funzione di controllo interno – la sua attivazione (valore pari a 1) permette di stampare a video le classi che sono state validate.

Analogamente al caso precedente, le classi che contengono un solo parametro vengono tralasciate per stabilità dei risultati.

Di seguito è riportata solamente la parte di codice in cui sono state effettuate le modifiche.

```

% AUTO CLASSIFICATION OF MODES BASED ON FREQUENCY AND MAC
% Class with just 1 mode are neglected for safety
debug = 0;
[; class_mode] = get_class_mode(FRE_IDmatSort, MAC_MATRIX);

```

7.1.3 *Get_class_modes*

Di seguito viene descritta la funzione che sta al cuore dell'automatizzazione del processo, che si basa sul concetto del *clustering c-means*, ovvero una tipologia di classificazione ‘dolce’ dei valori, e quindi che concettualmente si discosta dalla logica del filtraggio a setaccio usato invece nel controllo manuale, descritto al capitolo precedente.

L'algoritmo che è stato applicato è un *Fuzzy C-Means Clustering*, una tecnica scoperta negli anni '70 di clustering che si basa sul concetto che ogni dataset sia organizzato in gruppi di cluster per cui ogni punto appartiene a determinati cluster per un certo grado – ciò comporta la differenza più importante con i cosiddetti algoritmi con hard criteria, che al contrario stabiliscono l'appartenenza di un determinato dato a un solo dataset tra quelli individuati.[14]

La funzione FCM inizia con un'ipotesi iniziale casuale per l'individuazione dei centri dei cluster, che rappresentano la posizione media di ciascun cluster. Quindi, fcm assegna a ogni singolo dato – la frequenza nel caso in esame - un grado di appartenenza casuale a ciascun cluster, per affinare poi questo passaggio, aggiornando iterativamente i centri cluster e i gradi di appartenenza per ciascun dato. Alla fine fcm individua i centri cluster nella posizione corretta all'interno del dataset e, per ciascun punto, trova il grado di appartenenza al relativo cluster. Questa iterazione minimizza una *objective function*, che rappresenta la distanza tra il singolo valore e il centro del cluster più vicino, ponderata in base all'appartenenza di quel dato al cluster – questo perché l'algoritmo di fcm si basa su concetto che i singoli punti possano appartenere, con gradi diversi, a diversi cluster e non a uno soltanto, quindi in base al peso che si assegna a ogni cluster si può modificare in base ai valori di progetto la vicinanza di un singolo valore al cluster.

Il codice quindi analizza l'andamento dei dati, ricercando i valori medi, cioè i picchi, dei cluster e individua autonomamente dei criteri di similarità su cui basa poi l'assegnazione dei singoli valori al cluster a cui più si avvicinano – ciò permette quindi l'individuazione dei centri di ogni cluster e poi, nel caso in esame, l'assegnazione in base alla distanza dai picchi dei cluster di ogni singolo valori, cercando il valore minore, poiché in questo caso non è stato attribuito un peso diverso per ogni cluster.

La funzione inizia con la definizione dei valori che ottiene in input, in questo caso il vettore delle frequenze e la matrice dei MAC, le due variabili più importanti che definiscono meglio i cluster di cui si sta effettuando la ricerca.

```
function [mm, class_modes] = get_class_mode(freq, mac)
freq1 = freq;
```

Il primo passaggio riguarda la creazione di una matrice di dissimilarità, che si basa sul concetto che il valore pari all'unità del MAC corrisponde alla completa uguaglianza di due valori fra loro, per a mano a mano scendere, sempre dipendendo dalla relazione tra i due valori i cui indici sono il numero di riga e colonna che identifica la cella contenente il valore – quindi in questo caso, trattandosi dell'esatto opposto, la matrice di dissimilarità conterrà valori sempre compresi tra 0 e 1, però in questo caso 1 indica una dissonanza completa, mentre 0 l'uguaglianza.

Si inizializza quindi una matrice con dimensioni pari a quelle della matrice del MAC originaria composta da soli valori pari a 1, a cui poi si è scelto di sottrarre i valori della matrice originaria, una matrice di similarità in questo ragionamento, per ottenere quindi la matrice ricercata, con valori ovviamente compresi tra 0 e 1 – ci si aspetta che questa matrice sulla diagonale principale presenti valori pari a 0, poiché quella di partenza aveva valori pari a 1.

Il passaggio successivo è di grande importanza logica – infatti, per semplificare la matrice delle dissimilarità così ottenuta, tutti i valori che sono risultati non accettabili nel passaggio manuale (MAC minore 0.95, e quindi in questo caso i valori maggiori di 0.05) vengono arrotondati all'unità, in modo da essere poi scartati nei passaggi successivi.

```
MAC_one=ones(size(mac,1),size(mac,1));  
Mac_D = MAC_one - mac;  
mac_D(mac_D < 0.05) = 1;
```

Ora si passa all'analisi del vettore delle frequenze, la seconda variabile della funzione; viene subito inizializzata una matrice di dimensioni $n \times n$, dove n è la dimensione del vettore delle frequenze iniziale. Il ciclo for che segue ha come obiettivo il riempimento di tale matrice secondo le seguenti regole: qualora i valori dei due indici i e j fossero diversi – e quindi non ci si riferisca alla stessa cella, che comporranno poi la diagonale della matrice – la matrice creata salverà nella cella corrispondente il valore assoluto della differenza tra se stessa e il valore precedente.

Analogamente a ciò che accade per il calcolo del MAC, i valori che superano la soglia stabilita, 0.08 nel caso della frequenza, vengono arrotondati all'unità, per facilitarne l'eliminazione in seguito.

La necessità di creare qui una matrice di dissimilarità delle frequenze nasce per il passaggio successivo, che si vedrà trattare la somma delle dissimilarità delle due variabili, e ciò comporterebbe errore se si lasciasse la differenza delle frequenze come vettore.

```

FREQ_MATRIX=zeros(size(freq1,2),size(freq1,2));

for i=1:size(freq1,2)
for j=1:size(freq1,2)
if i ≠ j
FREQ_MATRIX(i,j)= abs((freq1(1,i)-freq1(1, j)));
end
end
end
FREQ_MATRIX(FREQ_MATRIX > 0.08) = 1;
disp(FREQ_MATRIX);

```

Per il calcolo quindi della matrice di dissimilarità da utilizzare poi per la ricerca dei cluster, non è rimasto che moltiplicare il peso per ogni valore ottenuto nelle singole matrici e quindi sommarle fra loro – da notare come ci si aspetta che i valori nelle celle siano compresi tutti tra 0 e 2, con necessità di scartare ovviamente tutti quelli maggiori di 0.13 (somma tra le soglie accettabili massime di $Mac = 0.05$ e $frequenza = 0.08$).

```

dissimilarity_matrix = (1.*mac_D)+(1.*FREQ_MATRIX);

```

La funzione *squareform* è un algoritmo di matlab con la finalità di convertire la matrice di dissimilarità appena trovata in un vettore quadrato, ovvero una matrice di distanza completa – è utile quando il calcolo comporta la creazione di una variabile contenente la distanza solo tra alcuni elementi ma per motivi algebrici è necessario avere la distanza rispettivamente tra tutti i punti.

```

VECTOR_DISSIM = squareform (dissimilarity_matrix);

```

Si passa poi alla funzione *linkage*, nativa in matlab, una funzione tipica degli algoritmi di clusterizzazione, appunto come l'FCM. In particolare viene molto utilizzata nella clusterizzazione gerarchica, ovvero nella classificazione di data set in base alla loro similarità o dissimilarità, per poi passare alla costruzione del dendrogramma che rappresenta i cluster e le loro interazioni, a partire da connessioni molto ampie e via via affinandosi.

Nello specifico in questo caso, la funzione *linkage* viene applicata al vettore quadrato appena trovato, che contiene quindi le dissimilarità tra elementi, secondo il principio average, ovvero il calcolo della distanza media di tutti i punti che appartengono al cluster, che permette alla fine di ottenere dei cluster compatti e bilanciati.

```

Z = linkage(VECTOR_DISSIM,'average');
% linkage(X,method) creates the tree using the specified method,
where method describes how to measure the distance between clusters.

```

Di seguito viene riportato un esempio di dendrogramma ricavato da questa parte di codice, in cui si nota come inizialmente i cluster considerati siano una parentesi molto ampia, per poi diminuire a ogni iterazione. Sull'asse delle ascisse sono riportati gli indici delle celle, quindi i valori di riferimento da clusterizzare, mentre sull'asse delle ordinate la distanza rispettiva tra punti.

Anticipando il passaggio successivo, è interessante sottolineare come l'impostazione del valore di cutoff a 0.13 (dato dalla somma delle soglie stabilite, 0.8 e 0.5) taglia via quasi ogni cluster ad eccezione di quello formato dalle celle con indici 3 e 4.

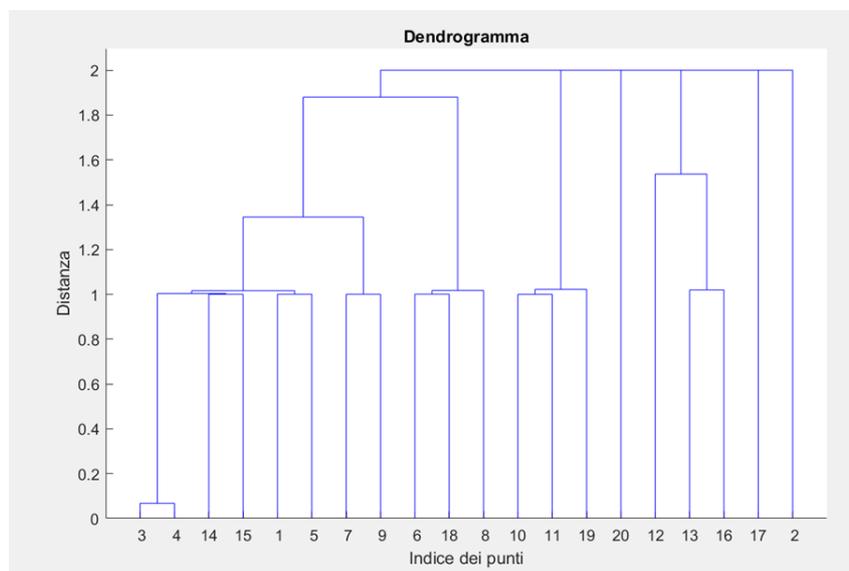


Figura 37: Esempio di un dendrogramma ricavato dall'FCM – grafico Z.

Il passaggio successive consiste nell'individuazione dei cluster partendo dai valori ricavati tramite il dendrogramma in figura. La funzione richiamata in matlab appunto ricava i cluster partendo dal grafico Z e usando come soglia di *cutoff* il valore di 0.13 – questo significa che vengono considerati validi i cluster che presentano distanza fra loro inferiore a questo valore, perchè come viene specificato tra gli altri parametri ci si basa sul concetto appunto di distanza minima.

```

% Extract indices of equal elements
equal_indices = cell(size(unique_elements));
for i = 1:length(unique_elements)
    equal_indices{i} = find(indices == i);
end

class_modes = equal_indices(cellfun(@numel, equal_indices) > 1);
valid = ones(1, length(class_modes));
mm = length(class_modes);

```

Come ultimo controllo, per evitare errori di displacement in altri punti del codice, è stato stabilito che i valori delle frequenze minori di 2.9 Hz e maggiori di 8 Hz non appartengono ai modi di vibrare consoni della struttura – sono infatti dati da per esempio le vibrazioni del ponteggio temporaneo che era stato montato durante una fase di registrazione dei dati, e quindi è stato deciso di scartarle, procedimento che è attuato dal ciclo for di seguito.

```

for i = 1:mm
    for j = 1:length(class_modes{i})
        idx = class_modes{i}(j);
        if(freq1(idx) < 2.9 || freq1(idx) > 8)
            valid(i) = 0;
            break;
        end
    end
end
end

```

Gli ultimi passaggi del codice sono semplicemente semplificazioni delle variabili, tra cui l'eliminazione dei cluster non validi e la trasposizione degli stessi per semplificare il salvataggio nella fase successiva, ma non sono di rilevanza per l'automatizzazione del procedimento.

```

Class_modes = class_modes(valid == 1);
Class_mode = ;
for i = 1:length(class_modes)
    class_mode{i} = class_modes{i}';
end
class_modes = class_mode;
mm = length(class_modes);

```

7.2 Salvataggio dei risultati e conclusione del processo automatico

Al termine dell'esecuzione del codice, i risultati ottenuti si presentavano nella forma di vettori di cluster, divisi in base alla cartella di riferimento iniziale della registrazione. Per raccogliere questi dati in un file più ordinato, si è deciso di unire la data e l'ora di riferimento del momento centrale del file originario, splittato in 20 minuti, la temperatura di riferimento di quel momento, e per concludere i risultati, sia sotto forma di valori medi della frequenza, forma modale e smorzamento, sia come variazione standard delle stesse variabili.

La conclusione più ovvia di questo processo è la riduzione drastica di tempo necessario per eseguire questa azione: infatti, in opposizione alla media di circa 5 minuti necessari per effettuare su ogni singolo file la codificazione manuale, il codice automatizzato ha impiegato circa 15 secondi, con una riduzione del tempo totale necessario del 95%, che in totale in questo caso quindi risulta essere di più di 190 ore.

Altro importante aspetto è l'eliminazione dell'errore umano dal procedimento, in quanto la ripetitività di un'azione crea distrazione per il tecnico, mentre l'utilizzo di un codice automatizzato riduce a zero questa variabile, notevole vantaggio.

Non si può negare il processo di trial-and-error necessario per raggiungere il risultato finale, che ha comportato quindi la scrittura di più parti del codice più volte, né la valutazione a più riprese della correttezza dei valori delle soglie individuate nel processo manuale, ma il risultato a cui si è giunti è stato ritenuto idoneo per lo scopo finale, anche se la brevità della registrazione, un lordo di tre mesi, con grandi lacune temporali in mezzo, non è stata di aiuto per la validazione completa dei risultati, che sono stati implementati grazie agli studi precedenti sulla stessa struttura.

7.3 Introduzione al *mode tracking*

A seguito del processo di identificazione e validazione dei cluster che vanno a individuare l'ordine del sistema e i modi di vibrare propri della struttura, si è predisposto il terreno per effettuare il processo di *mode tracking*, un procedimento di tracciamento dei modi, per correlare stessi modi da registrazioni o modelli differenti.

È proprio in questo procedimento che il lavoro di automatizzazione del processo validativo dei modi di vibrare identificati presentato in questo elaborato vede il futuro prossimo –

è un procedimento di fondamentale importanza, che, se effettuato erroneamente, quindi ad esempio assegnando erroneamente un file a una determinata sequenza storica, può causare considerazioni sbagliate sullo stato di danneggiamento della struttura monitorata.

Nel caso studio della chiesa di Santa Caterina, è inoltre essenziale procedere con la massima attenzione in questo passaggio, dal momento che, essendo la rete di sensori applicata molto scarsa (totale due sensori, su tre direzioni di registrazione), il parametro di correlazione dei diversi modi di vibrare, ad esempio il MAC, *Modal Assurance Criterion*, perde di robustezza, effetto che causa un drastico aumento della possibilità di errore nella correlazione appunto tra modi provenienti da identificazioni diverse.

Per scongiurare quindi la possibilità che avvengano questo tipo di errori, causati dalla ridotta robustezza dei parametri, si può introdurre un valore minimo di soglia accettabile del parametro di riferimento, come il MAC appunto, al di sotto del quale i valori provenienti dall'identificazione verranno semplicemente scartati.

Il valore della soglia è stimato, e variato durante il procedimento se necessario, tenendo in considerazione diversi fattori, come il numero di outliers presenti (ovvero valori, solitamente puntuali, che si discostano evidentemente dal trend dei dati) e il numero di valori scartati, quindi valori nulli, ma anche tenendo presente l'andamento delle registrazioni di frequenza, se oscillante attorno a un solo valore, o presente su un range più elevato di valori, magari creando anche trend differenti – dall'equilibrio di tutti questi parametri si giunge ai valori ottimali di soglia, secondo decisioni progettuali.[11]

Come dichiarato in precedenza, la scarsità dei sensori utilizzati nel caso studio e la brevità del periodo di registrazione a lungo termine (3 mesi non completi, intervallati da lacune anche di diverse settimane) hanno reso il processo di *mode tracking* più complesso, a causa della mancanza di robustezza dei parametri di correlazione; ciò ha portato alla necessità di analisi più attente e approfondite del procedimento, il quale quindi non ha trovato spazio nel presente elaborato.

8 Conclusione

Questo elaborato si inserisce nel grande ombrello della letteratura concerne lo *Structural Health Monitoring*, una tecnica sviluppata in campo ingegneristico a partire dagli anni '80 e che, grazie anche al supporto della tecnologia sempre crescente, sta vedendo un grande sviluppo, anche in virtù dei vantaggi che la sua applicazione comporta, se confrontati con altri metodi di identificazione del danno. In particolare la non invasività e non distruttività, lo rendono il metodo migliore di analisi delle condizioni strutturali di edifici monumentali, come appunto la Chiesa di Santa Caterina, oggetto di studio e monitoraggio da più di 10 anni, in maniera non continuativa.

L'obiettivo di questo elaborato era lo studio del processo di *data analysis*, dal preprocessing dei dati, passando per l'identificazione e infine il postprocessing, la parte che ha richiesto una maggiore attenzione, poiché risultava essere l'unico passaggio per cui ancora non era stato validato un procedimento automatico.

I dati, raccolti dal sistema di monitoraggio a lungo termine a due sensori, con tre canali di registrazione, per circa tre mesi, sono stati prima divisi in file dalla medesima lunghezza, per poter essere confrontati tra loro, poi preprocessati, in maniera automatica, in modo da eliminare disturbi e rumore sulle registrazioni.

La parte successiva riguardava l'identificazione dei modi strutturali e dell'ordine del sistema, tramite la computazione con un codice già testato e approvato per lo stesso scopo sul santuario di Vicoforte (Epicoco 2022) – l'algoritmo qui applicato è del tipo *linkage clustering*.

Il cuore dell'elaborato è costituito invece dalla parte di postprocessing dei dati, in cui vengono elaborati i risultati identificati per essere raggruppati e classificati come modi propri del sistema, scartando i valori che rappresentano modi solamente numerici. Questa parte è stata attuata prima in maniera manuale su una parte dei file, in particolare quelli attorno ai picchi locali di temperatura, per permettere di elaborare il processo logico da seguire al fine di automatizzare la procedura e confrontare i risultati.

Per l'automatizzazione è stata cruciale la scelta della tipologia di algoritmo da utilizzare, che è ricaduta sul *Fuzzy C-Means*, un algoritmo di clusterizzazione, che concettualmente raggruppa i dati in base alla similarità tra loro, ovvero si basa sulla vicinanza dei parametri

di controllo, come la frequenza e il MAC dei singoli modi, e che quindi porta a scartare valori che superano una determinata soglia di dissimilarità e ad approvare quelli che la rispettano. L'elaborazione successiva trattava il calcolo della media dei parametri sui cluster individuati, per ottenere i valori finali caratterizzanti i modi naturali della struttura, che sono anche stati confrontati con quelli già individuati durante campagne di monitoraggio precedenti della medesima costruzione e hanno presentato una coerenza accettabile, per cui è stato ritenuto valido il processo.

Sono state quindi preposte le basi per un successivo procedimento di *mode tracking*, ovvero l'assegnazione dei risultati ottenuti ai modi correlati individuati in registrazioni differenti o in modelli precedenti. Questo procedimento potrebbe nascondere qualche insidia dovuta alla mancanza di robustezza dei dati ottenuti, a causa del ridotto numero di sensori.

Il presente elaborato pone quindi si colloca nel processo di automatizzazione dell'analisi dati proveniente dai sistemi di rilevazione posti in situ, che ha quindi l'obiettivo di velocizzare le procedure per rendere più agevole il lavoro e più tempestivo l'intervento in caso di necessità strutturale.

Riferimenti bibliografici

- [1] “<https://www.comune.casale-monferrato.al.it/chiesa-santa-caterina>.” Consultato 17 ottobre 2023.
- [2] “<https://www.santacaterinacasalemonferrato.it/>.” Consultato 20 settembre 2023.
- [3] M. Di Carlo, *Analisi e progetto del miglioramento sismico della facciata della chiesa barocca di Santa Caterina a Casale Monferrato = Analysis and assessment for the seismic retrofit of the facade of the baroque church of Santa Caterina in Casale Monferrato*. PhD thesis, Politecnico di Torino, 2022.
- [4] C. R. Farrar and K. Worden, “An introduction to structural health monitoring,” *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 365, no. 1851, pp. 303–315, 2007.
- [5] R. Moss and S. Matthews, “In-service structural monitoring. a state of the art review,” *Structural Engineer*, vol. 73, no. 2, 1995.
- [6] A. Mita, “Emerging needs in japan for health monitoring technologies in civil and building structures,” in *Proc. Second International Workshop on Structural Health Monitoring*, pp. 56–67, Citeseer, 1999.
- [7] J. M. Brownjohn, “Structural health monitoring of civil infrastructure,” *Philosophical Transactions of the Royal Society A: Mathematical, Physical and Engineering Sciences*, vol. 365, no. 1851, pp. 589–622, 2007.
- [8] K. Worden and J. M. Dulieu-Barton, “An overview of intelligent fault detection in systems and structures,” *Structural Health Monitoring*, vol. 3, no. 1, pp. 85–98, 2004.
- [9] A. Rytter, “Vibrational based inspection of civil engineering structures,” 1993.
- [10] R. Epicoco, *Identificazione modale automatica di tipo Stochastic Subspace per il monitoraggio di grandi edifici monumentali = Automatic modal identification with Stochastic Subspace for monumental structures monitoring*. PhD thesis, Politecnico di Torino, 2022.
- [11] S. Coccimiglio, G. Miraglia, G. Coletta, R. Epicoco, and R. Ceravolo, “Balanced definition of thresholds for mode-tracking in a long-term seismic monitoring system,” 2023.
- [12] B. Peeters and G. De Roeck, “Reference based stochastic subspace identification in civil engineering,” *Inverse problems in Engineering*, vol. 8, no. 1, pp. 47–74, 2000.
- [13] R. Brincker and P. Andersen, “Understanding stochastic subspace identification,” in *Conference Proceedings: IMAC-XXIV: A Conference & Exposition on Structural Dynamics*, Society for Experimental Mechanics, 2006.

[14] “<https://it.mathworks.com/help/fuzzy/fuzzy-c-means-clustering.html>.” Consultato 2 ottobre 2023.

Ringraziamenti

È doveroso porgere i miei ringraziamenti alle figure accademiche che hanno reso possibile la redazione di questo elaborato: in primis il professor Rosario Ceravolo, per l'opportunità offertami in questo momento fondamentale della mia carriera universitaria; a tutti i corelatori, che, dall'inizio in cui tutto appariva appannato, hanno contribuito a diradare la nebbia, hanno risposto ai miei dubbi e hanno corretto i miei errori, in particolare ci terrei a ringraziare il Ph.D. Gaetano Miraglia e l'ing. Stefania Coccimiglio, per il grande supporto lungo tutto il percorso.

Grazie alla mia famiglia, il porto sicuro che è stato scosso da una terribile tempesta negli ultimi anni.

A mamma, sei il mio esempio di vita e di forza.

A Emily, perchè anche se forse non ci teniamo sempre per mano, saremo sempre una accanto all'altra.

Un pensiero speciale a papà, questo traguardo significava tanto per te, non ti vedrò oggi sorridere davanti a me, ma sono sicura di averti reso fiero, ovunque tu sia.

Grazie a Michele, perché quando tutto è buio e confuso, tu accendi la luce – sei speciale.

Grazie a tutti i miei amici, a chi c'è da sempre e a chi si è aggiunto da poco – siete le stelle che fanno brillare la vita.