Designing Health Indicators for Aerospace Structures by Intelligent Information Fusion

Moradi, M.

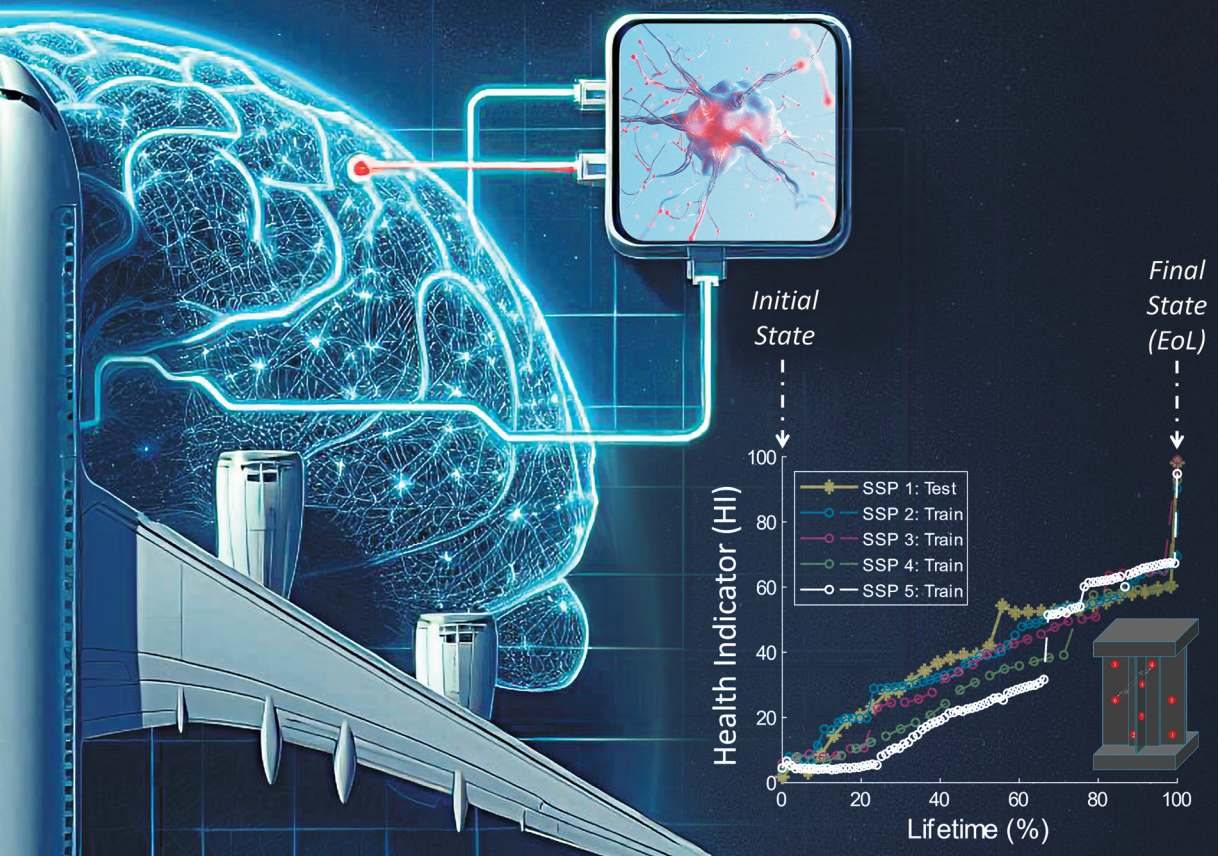
DOI
10.4233/uuid:7ac03701-b97a-427d-990c-e6c696d1254b

Publication date
2024

Document Version
Final published version

Citation (APA)
Moradi, M. (2024). *Designing Health Indicators for Aerospace Structures by Intelligent Information Fusion*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:7ac03701-b97a-427d-990c-e6c696d1254b

Designing Health Indicators for Aerospace Structures by Intelligent Information Fusion



Morteza Moradi

Propositions

accompanying the dissertation

Designing Health Indicators for Aerospace Structures by Intelligent Information Fusion

by

Morteza Moradi

1. The primary outcome of your PhD is not your thesis (output); it is you (the trained model).
2. Failures teach (backpropagation), and successes are the fruits of failures. While the latter (output) excites you temporarily, the former (learning ability) develops skills that last a lifetime.
3. Initial weights and biases have unavoidably an impact on a deep learning model's performance, much like the effect of where and when people are born in their lives. Nevertheless, robust models must always perform well, regardless of their initial condition. [Generalizability]
4. Occupying a position in a prestigious institute does not mean that you are good (criteria given solely for the test unit); rather, some of your predecessors were good, making the institute prestigious (criteria given for all units, including training and test units). [Thesis]
5. When we claim we comprehend how a complex engineering system operates, then either we are dishonest or the system is not complex. [Interpretability] [Thesis]
6. Unlimited resources may blind you to opportunities, whereas confronting constraints serves to broaden your perspective. To flourish within constraints, you have to improve your skills and become stronger.
7. There is no logical basis to assume that the true remaining useful life is piece-wise linear.
8. Competition is an essential ingredient of development, regardless of whether the effort is individual or collaborative.
9. Striving to meet all criteria to an acceptable standard is more advantageous than achieving high scores in some while neglecting others. [Thesis]
10. Reading a paper costs; writing one costs; and reviewers are not also paid. It has the ingredients of a belief system that no one questions.

These propositions are regarded as opposable and defendable, and have been approved as such by the promotor prof. dr. ir. R. Benedictus and copromotor dr. D. Zarouchas.

Designing Health Indicators for Aerospace Structures by Intelligent Information Fusion

Designing Health Indicators for Aerospace Structures by Intelligent Information Fusion

Dissertation

for the purpose of obtaining the degree of doctor
at Delft University of Technology,
by the authority of the Rector Magnificus Prof.dr.ir. T.H.J.J. van der Hagen,
chair of the Board of Doctorates,
to be defended publicly on
Monday, 16^{th} September 2024 at 10:00 o'clock

by

Morteza Moradi

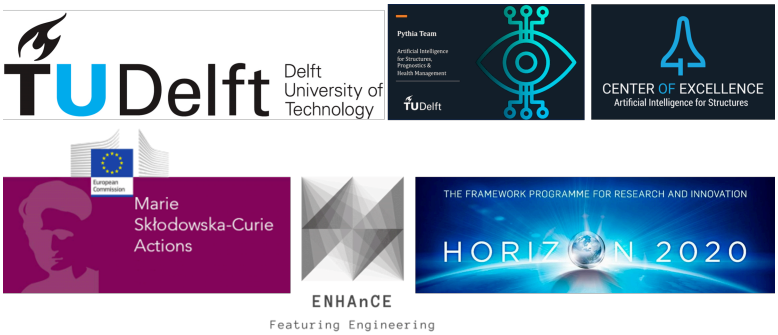
Master of Science in Aerospace Engineering
Iran University of Science & Technology
born in Isfahan, Iran.

This dissertation has been approved by the promotors.

Composition of the doctoral committee:

Rector Magnificus,	Chairperson
Prof. dr. ir. R. Benedictus,	Delft University of Technology, promotor
Dr. D. Zarouchas,	Delft University of Technology, promotor
Independent members:	
Prof. dr. S.J. Watson,	Delft University of Technology
Prof. dr. N. Dervilis,	University of Sheffield, UK
Prof. dr. Z. Sharif Khodaei,	Imperial College London, UK
Prof. dr. M. Kotsonis,	Delft University of Technology
Dr. T. Loutas,	University of Patras, Greece
Dr. ir. R. De Breuker,	Delft University of Technology, reserve member

The thesis was conducted under the European Union's Horizon 2020 program, supported by Marie Sklodowska-Curie grant No. 859957, titled "ENHAnCE: European training Network in intelligent prognostics and Health mAnagement in Composite structurEs".



Keywords: Structural health monitoring; Prognostics; Artificial intelligence; Information fusion; intelligent health indicator; Semi-supervised learning; Interpretable neural network; Composite structures; Acoustic emission; Guided waves; Data-driven models

Printed by: Ipskamp printing

Front & Back: Cover design by P. Raeis Ghanavati, with AI-generated elements.

ISBN (E-book) 978-94-6384-631-8
ISBN (Paperback) 978-94-6384-632-5

An electronic version of this dissertation is available at
http://repository.tudelft.nl/.

"درخت تو گر بار دانش بگیرد به زیر آوری چرخ نیلوفری را"

ناصر خسرو قبادیانی، شاعر و فیلسوف ایرانی قرن یازدهم

"Interpretation: If you gain knowledge and wisdom, you can even control the sky (the passage of time)."

Nasir Khusraw, 11th-century Persian poet and philosopher

Summary

Awareness of the health status of an engineering system holds significant importance across multiple industries, such as aerospace, marine, and energy. This awareness can take two primary forms—either detailed or comprehensive—each carrying its own distinct benefits. However, when it comes to pivotal decisions concerning the prediction of remaining useful life (RUL), the initiation of shutdown procedures, and the identification of maintenance requirements, the need for a comprehensive health indicator (HI) becomes essential. Such a HI serves as a vital metric for predicting RUL, facilitating maintenance decisions, and refining structural designs.

Composite structures play a pivotal role in aerospace and wind energy applications, offering high performance, lightweight construction, and durability, which result in safety enhancements and cost savings. However, unlike conventional structures, they are susceptible to various forms of damage, such as matrix cracking, delamination, and fiber buckling, leading to diverse failure scenarios. The HI should represent all these failure modes, acting as an intermediary that connects data from structural health monitoring (SHM) techniques with prognostic models for predicting RUL. Despite advancements in understanding material deformation and fracture mechanisms through defect theory and crack mechanics, predicting failures in composite structures at a macroscopic level remains challenging.

Highlighting the significance of HI as a macroscopic index for predicting RUL and guiding maintenance decisions, this thesis aims to develop intelligent frameworks for HI extraction tailored for aerospace composite structures. The literature review identifies gaps in existing methodologies, including: limited applicability to composite structures; the need for nonlinear HIs aligned with damage propagation physics to move beyond the conventional linear assumptions; and the importance of generalizability and criteria evaluation, with a focus on test units. Additionally, the need for historical-independent HI, addressing the common historical-dependency drawback in HI construction and prognostic models; the fusion of passive and active SHM techniques to maximize the benefits of both temporal and spatial information; and emphasizing the interpretability of a HI are highlighted.

A new methodology is introduced for developing HIs, emphasizing the importance of nonlinearity aligned with damage propagation physics. Prognostic criteria for evaluating HIs are highlighted and refined in terms of model stability and performance on test units. The developed methodology integrates a nonlinear kernel within a semi-supervised learning paradigm, serving as the foundation for subsequent frameworks designed to address the other above-mentioned research gaps and specific case studies. To validate the proposed methodology, different experimental data were employed, where two SHM techniques—acoustic emission (AE) and guided wave (GW)—were utilized to monitor composite structures.

The primary experimental campaign focused on T-stiffener composite panels subjected to impact and compression-compression fatigue loadings (ReMAP dataset), highlighting

the challenges of realistic and uncertain phenomena during experiments, resulting in a wide range of end-of-life (EoL). Stiffened composite panels were monitored using various SHM techniques, with AE and GW being the primary focus. Three data-driven frameworks, leveraging signal processing and artificial intelligence (AI), were developed based on AE data to construct HIs. Twelve units were employed for training, validation, and testing, using leave-one-out cross-validation (LOOCV) to rigorously evaluate generalizability.

Regarding the GW monitoring technique, a data-driven framework employing signal processing and deep learning was developed. Two distinct datasets—ReMAP and NASA—were investigated, featuring stiffened composite panels and dogbone specimens subjected to fatigue loading. The ReMAP dataset comprises five T-stiffener composite units monitored by the GW technique, while the NASA dataset involves tension-tension fatigue loading on dogbone composite specimens with three different layups (4, 3, and 4 units for each, respectively). The LOOCV process was used to evaluate the developed framework on each dataset. Given the targeted GW datasets, the proposed GW-based framework should be flexible in accommodating various sensor numbers, networks, excitation frequencies, and setups. Moreover, the GW technique, as an active SHM technology, was selected as a good candidate for exclusively utilizing current monitored data to address the historical dependency drawback.

Furthermore, another key aspect of the thesis involves the fusion of information from different SHM techniques. The ReMAP dataset was considered for the GW-AE fusion scenario, where data from both AE and GW were integrated to formulate HIs for T-stiffener composite panels. To showcase the effectiveness of the data-driven fusion approach in improving the overall performance of SHM techniques, the LOOCV process was employed.

One aspect that contributes to interpretability is the presence of incremental steps in the generated HIs, which may correspond to distinct damage states within composite structures. These incremental steps can offer valuable insights for informing state-based RUL prediction models. To address this, AE- and GW-based frameworks were designed and investigated. Additionally, the development of an interpretable neural network (INN) was aimed at designing HIs for commercial turbofan engines using the CMAPSS dataset. The goal was to achieve optimal performance by condensing overly complex equations into a more readable format. To achieve this, multiplicative neurons were developed to complement additive neurons, with sparsity control applied to discretize weights. Despite the shift in focus from composite structures to turbofan engines, which presents a more straightforward dataset compared to those involving composite structures, INN successfully provides an interpretable HI for the CMAPSS dataset. This outcome further validates the effectiveness of the semi-supervised method in extracting HIs.

Overall, the thesis contributes to advancing prognostics and health management (PHM) in aerospace engineering by providing innovative methodologies and insights for designing HIs tailored for composite structures. The research findings underscore the importance of addressing challenges such as complexity, interpretability, historical data dependency, and the fusion of SHM techniques, laying the groundwork for future advancements in the field.

Samenvatting

Bewustzijn van de gezondheidstoestand van een technisch systeem heeft aanzienlijk belang binnen meerdere industrieën, zoals de luchtvaart, maritieme, en energie-industrieën. Dit bewustzijn kan twee primaire vormen aannemen—gedetailleerd of uitgebreid—die elk hun eigen specifieke voordelen met zich meebrengen. Echter, wanneer het aankomt op cruciale beslissingen met betrekking tot de voorspelling van de resterende bruikbare levensduur (RUL), het starten van shutdown-procedures, en het identificeren van onderhoudsvereisten, wordt de behoefte aan een uitgebreide gezondheidsindicator (HI) essentieel. Zo'n HI fungeert als een vitaal meetinstrument voor het voorspellen van RUL, het vergemakkelijken van onderhoudsbeslissingen, en het verfijnen van structurele ontwerpen.

Composiet structuren spelen een cruciale rol in luchtvaart- en windenergietoepassingen, waarbij ze hoge prestaties, lichtgewicht constructie, en duurzaamheid bieden, wat resulteert in verbeterde veiligheid en kostenbesparingen. Echter, in tegenstelling tot conventionele structuren, zijn ze vatbaar voor verschillende vormen van schade, zoals matrixscheuren, delaminatie, en vezel knik, wat leidt tot diverse faalscenario's. De HI dient al deze faalmodi te omvatten, fungerend als een tussenstap die gegevens van structurele gezondheidsmonitoring (SHM) technieken verbindt met prognosemodellen voor het voorspellen van RUL. Ondanks vooruitgang in het begrip van materiaaldeformatie en breukmechanismen via defectentheorie en mechanica, blijft het voorspellen van fouten in composiet structuren op macroscopisch niveau uitdagend.

Met de nadruk op de betekenis van HI als macroscopische index voor het voorspellen van RUL en het begeleiden van onderhoudsbeslissingen, heeft deze thesis tot doel intelligente kaders te ontwikkelen voor HI-extractie op maat voor luchtvaartcomposietstructuren. De literatuurstudie identificeert kloven in bestaande methodologieën, waaronder: beperkte toepasbaarheid op composietstructuren; de behoefte aan niet-lineaire HI's in overeenstemming met de fysica van schadepropagatie om voorbij de conventionele lineaire aannames te gaan; en het belang van generaliseerbaarheid en criteria-evaluatie, met de nadruk op testeenheden. Bovendien worden de behoefte aan geschiedenisonafhankelijke HI, het aanpakken van het gemeenschappelijke geschiedenis afhankelijkheidsnadeel in HI-constructie en prognosemodellen; de fusie van passieve en actieve SHM-technieken om de voordelen van zowel tijdelijke als ruimtelijke informatie te maximaliseren; en het benadrukken van de interpretatiemogelijkheden van een HI zijn belicht.

Een nieuwe methodologie wordt geïntroduceerd voor het ontwikkelen van HI's, waarbij de nadruk wordt gelegd op het belang van niet-lineariteit in overeenstemming met de fysica van schadepropagatie. Prognostische criteria voor het evalueren van HI's worden benadrukt en verfijnd qua modelstabiliteit en prestaties op testeenheden. De ontwikkelde methodologie integreert een niet-lineaire kernel binnen een semi-supervised learning-paradigma, als basis voor daaropvolgende kaders die zijn ontworpen om de andere bovengenoemde onderzoekskloven en specifieke casestudies aan te pakken.

Om de voorgestelde methodologie te valideren, werden verschillende experimentele gegevens gebruikt, waarbij twee SHM-technieken—akoestische emissie (AE) en geleide golven (GW)—werden gebruikt om composiet structuren te monitoren.

De primaire experimentele campagne richtte zich op T-stijve composietpanelen die werden blootgesteld aan inslag- en compressie-compressie-vermoeidheidsbelastingen (ReMAP-dataset), waarbij de uitdagingen van realistische en onzekere verschijnselen tijdens de experimenten werden benadrukt, wat resulteerde in een breed scala aan einde van het leven (EoL). Verstijfde composietpanelen werden gemonitord met verschillende SHM-technieken, waarbij AE en GW de belangrijkste focus waren. Drie op gegevens gebaseerde kaders, waarbij signaalverwerking en kunstmatige intelligentie (AI) werden ingezet, werden ontwikkeld op basis van AE-gegevens om HI's te construeren. Twaalf eenheden werden gebruikt voor training, validatie en testen, waarbij leave-one-out cross-validatie (LOOCV) werd toegepast om de generaliseerbaarheid grondig te evalueren.

Met betrekking tot de GW-monitoringtechniek werd een op-gegevens-gebaseerd kader ontwikkeld waarbij signaalverwerking en diep leren werden toegepast. Twee verschillende datasets—ReMAP en NASA—werden onderzocht, met verstijfde composietpanelen en proefstuk met verjonging die aan vermoeidheidsbelasting werden onderworpen. De ReMAP-dataset omvat vijf T-stijve composiete eenheden die door de GW-techniek worden gemonitord, terwijl de NASA-dataset spannings-spanningsvermoeidheidstests op composietspecimens (met verjonging) met drie verschillende legpatronen omvat (respectievelijk 4, 3, en 4 eenheden). Het LOOCV-proces werd gebruikt om het ontwikkelde kader op elke dataset te evalueren. Gezien de gerichte GW-datasets, zou het voorgestelde op GW gebaseerde kader flexibel moeten zijn in het omgaan met verschillende sensoraantallen, netwerken, excitatiefrequenties, en opstellingen. Bovendien werd de GW-techniek, als een actieve SHM-technologie, geselecteerd als een goede kandidaat voor het uitsluitend gebruiken van live gemonitorde gegevens om het geschiedenis-afhankelijkheidsnadeel aan te pakken.

Verder omvat een ander sleutelaspect van de thesis de fusie van informatie uit verschillende SHM-technieken. De ReMAP-dataset werd overwogen voor het GW-AE fusiescenario, waarbij gegevens van zowel AE als GW werden geïntegreerd om HI's te formuleren voor T-stijve composietpanelen. Om de effectiviteit van de op gegevens gebaseerde fusiebenadering in het verbeteren van de algehele prestaties van SHM-technieken te laten zien, werd het LOOCV-proces toegepast.

Een aspect dat bijdraagt aan de interpreteerbaarheid is de aanwezigheid van incrementele stappen in de gegenereerde HI's, die kunnen overeenkomen met verschillende schadetoestanden binnen composiet structuren. Deze incrementele stappen kunnen waardevolle inzichten bieden voor het informeren van op staten gebaseerde RUL-voorspellingsmodellen. Om dit aan te pakken, werden AE- en GW-gebaseerde kaders ontworpen en onderzocht. Bovendien was het ontwikkelen van een interpreteerbaar neuraal netwerk (INN) gericht op het ontwerpen van HI's voor commerciële turbofanmotoren met behulp van de CMAPSS-dataset. Het doel was om optimale prestaties te behalen door overdreven complexe vergelijkingen om te zetten in een leesbaarder formaat. Om dit te bereiken, werden multiplicatieve neuronen ontwikkeld om additieve neuronen aan te vullen, met toepassing van spaarzaamheidscontrole om gewichten te discretiseren. Ondanks de verschuiving van de focus van samengestelde structuren naar turbofanmotoren,

wat een eenvoudiger dataset oplevert vergeleken met die welke samengestelde structuren omvatten, biedt INN met succes een interpreteerbare HI voor de CMAPSS-dataset. Dit resultaat valideert verder de effectiviteit van de semi-supervised methode in het extraheren van HI's.

Al met al draagt de thesis bij aan het bevorderen van prognostiek en gezondheidsmanagement (PHM) in de luchtvaarttechniek door innovatieve methodologieën en inzichten te bieden voor het ontwerpen van HI's op maat voor composietstructuren. De onderzoeksresultaten benadrukken het belang van het aanpakken van uitdagingen zoals complexiteit, interpreteerbaarheid, geschiedenis-gegevensafhankelijkheid, en de fusie van SHM-technieken, wat de basis legt voor toekomstige vooruitgang op dit gebied.

Contents

Nomenclature

Abbreviations

AdaBoost	Adaptive Boosting
AE	Acoustic Emission
AI	Artificial Intelligence
ANN	Artificial Neural Network
BiLSTM	Bidirectional Long Short-Term Memory
BiLSTM-EL	Bidirectional Long Short-Term Memory Ensemble Learner
BO	Bayesian Optimization
C-C	Compression-Compression
CAE	Contractive Autoencoder
CBM	Condition-Based Maintenance
CEEMDAN	Complete Ensemble Empirical Mode Decomposition with Adaptive Noise
CEEMDAN-SSEDL	Semi-Supervised Ensemble Deep Learning (fed by intrinsic mode functions of CEEMDAN)
CFRP	Carbon Fiber-Reinforced Polymer
CM	Condition Monitoring
CNN	Convolutional Neural Network
CRNN	Convolutional Recurrent Neural Network
DBN	Deep Belief Network
DFOS	Distributed Fiber Optic Sensing
DFT	Discrete Fourier Transform
DI	Damage Index
DIC	Digital Image Correlation
DL	Deep Learning
DNN	Deep Neural Network
DPCA	Dynamic Principal Component Analysis
EEMD	Ensemble Empirical Mode Decomposition
EL	Ensemble Learning
ELM	Ensemble Learning Model
EMD	Empirical Mode Decomposition

EoL	End-of-Life
ESA	Ensemble Stacked Autoencoder
FBG	Fiber Bragg Grating
FC	Fully Connected
FC-EL	Fully Connected Ensemble Learner
FE	Feature Extraction
FF	Feature Fusion
FFT	Fast Fourier Transform
FFT-PCA-2S-ML-PBO	Two-Stage Machine Learning Approach with Physics-based Bayesian Optimization (following PCA-based dimension reduction upon time's and frequency's features
FFT-SSLSTM	Semi-Supervised Long Short-Term Memory (fed by time's and frequency's features)
FIR	Finite Impulse Response
GP	Genetic Programming
GPR	Gaussian Process Regression
GW	Guided Waves
HD	Historical Dependence
HI	Health Indicator
HT	Hilbert Transform
HT-SSCNN	Hilbert Transform-Semi-Supervised Convolutional Neural Network
IMF	Intrinsic Mode Function
IML	Interpretable Machine Learning
INN	Interpretable Neural Network
IRT	Infrared Thermography
JDL	Joint Directors of Laboratories
KPCA	Kernel Principal Component Analysis
LOOCV	Leave-One-Out Cross-Validation
LR_R	Robust Linear Regression using the bisquare weight function
LR_S	Linear model using Stepwise Regression
LSBoost	Least-Squares Boosting
LSTM	Long Short-Term Memory
LW	Lamb Wave
MK	Mann-Kendall
ML	Machine Learning
MLP	MultiLayer Perceptron

MMAE	Modified Mean Absolute Error
MMK	Modified Mann-Kendall
Mo	Monotonicity
MQE	Minimum Quantization Error
MSE	Mean-Squared Error
MWPCA	Moving-Window Principal Component Analysis
NDE	Non-Destructive Evaluation
NDT	Non-Destructive Testing
NDTE	Non-Destructive Testing and Evaluation
NLP	Natural Language Processing
PBO	Physics-based Bayesian Optimization
PC	Principal Component
PCA	Principal Component Analysis
PHI	Physical Health Indicator
PHM	Prognostics and Health Management
PP	Pre-Processing
Pr	Prognosability
RA	Rise-time/Amplitude ratio
ReLU	Rectified Linear Units
RMS	Root Mean Square
RMSE	Root Mean Square Error
RNN	Recurrent Neural Network
RNN-HI	Recurrent Neural Network-based Health Indicator
RPCA	Recursive Principal Component Analysis
RRMS	Relative Root Mean Square
RUL	Remaining Useful Life
SAE	Simple Averaging Ensemble
seq2seq	Sequence-to-Sequence
SHM	Structural Health Monitoring
SNR	Signal-to-Noise Ratio
SOM	Self-Organizing Map
SOM-HI	Self-Organizing Map-based Health Indicator
SP	Signal Processing
SS	Semi-Supervised
SSAE	Stacked Sparse Autoencoder

SSC	Semi-Supervised Classification
SSCNN	Semi-Supervised Convolutional Neural Network
SSDNN	Semi-Supervised Deep Neural Network
SSL	Semi-Supervised Learning
SSLSTM	Semi-Supervised Long Short-Term Memory
SSP	Single-Stiffened composite Panel
SSR	Semi-Supervised Regression
SVM	Support Vector Machine
T-T	Tension-Tension
TDM	Historical- or Time-Dependent Model
TIM	Historical- or Time-Independent Model
Tr	Trendability
Tree	Binary Decision Tree
VAE	Variational Autoencoder
VHI	Virtual Health Indicator
WAE	Weighted Averaging Ensemble
XAI	eXplainable Artificial Intelligence
XGB	eXtreme Gradient Boosting

Symbols

$*$	The convolution operation
δHI	Relative health indicator
δRUL	Relative remaining useful life
η	Forgetting factor in RPCA
$\hat{x}(t)$	Hilbert transform for signal $x(t)$
λ	Regularization parameter
μ	Mean value
$\omega_k^{Fitness\_MSE}$	Weight of the k^{th} individual base learner model based on the combination of Fitness and MSE metrics
$\omega_k^{Fitness}$	Weight of the k^{th} individual base learner model based on the Fitness metric
ω_k^{MSE}	Weight of the k^{th} individual base learner model based on the MSE metric
ω_k	Weight of the k^{th} individual base learner model
σ_{x_j}	The standard deviation of x_j
σ_{x_k}	The standard deviation of x_k
τ'	All units set except for the test ones
τ	The test units set

b^l	The bias relevant to the l^{th} layer
$cov(x_j, x_k)$	Covariance of x_j and x_k
D	Training set
$E_C(y, \hat{y})$	The mean-squared loss function between labels y and predicted outputs $\hat{y}$
$E_R(w')$	The discretizing loss
E	Loss function
$F(N)$	Activation function on the neuron N
f_k	The k^{th} individual base learner model
f_{WAE}	The weighted averaging ensemble learner model
$h(f)$	The Hilbert envelope spectrum for signal $x(t)$
ln	Natural logarithm
L	Number of layers
$M^{\tau'}$	Number of all units set except for the test ones
M^{τ}	Number of test units
Mo	Monotonicity
M	Number of units (specimens)
N_j	Number of measurements for the jth unit
N_k	Number of measurements for the kth unit
Pr	Prognosability
$sgn(\bullet)$	Signum function
t_0	The operational time in terms of cycles at the beginning
T_i	The target value (simulated HI under SSL paradigm) for time step i
t_{EoL}	The operational time in terms of cycles at the EoL
Tr	Trendability
t	Operating time
v	Validation set
w'	Constrained weights
w_{ji}^l	The weight relevant to the connection between the j^{th} neuron at the l^{th} layer to the $(l-1)^{th}$ layer's i^{th} neuron
$x(t_i)$	Measurements (HIs in the context) at the times of t_i
$x(t_p)$	Measurements (HIs in the context) at the times of t_p
x^{ast}	Adaptively standardized x by the adaptive zero-mean normalization
x^{norm}	Normalized x by the min-max normalization
x^{st}	Standardized x by the zero-mean normalization
$x_j(1)$	The HI value of the jth unit at the initiation

$x_j\left(N_j\right)$	The HI value of the jth unit at the EoL
x_j	Vectors of measurements for the jth unit
x_k	Vectors of measurements for the kth unit
X	The entire historical data from the beginning to the EoL
$z(t)$	The analytic signal for signal $x(t)$
$\bar{\omega}_k$	Normalized influential weight of the k^{th} individual base learner model
$Fitness^{\tau}$	Fitness given test units
HI_i	The network's output for time step i
Mo^{τ}	Monotonicity given test units
Pr^{τ}	Prognosability given test units

List of Figures

List of Tables

1

Introduction

In this introductory chapter, the definitions and motivation for a health indicator in comparison to the remaining useful life of an engineering system are elucidated, and the essential research objectives regarding the construction of health indicators are expounded upon.

1

1.1 Health Indicator & Remaining Useful Life

Awareness of an engineering system's health status is paramount in a range of industries, including aerospace, marine, and energy. This awareness can manifest in two forms—detailed or comprehensive—each offering its own set of advantages. For instance, acquiring detailed knowledge of the composite structure's damage—such as microcracks, cracks, fiber breakages, voids, and delaminations—proves invaluable for delving into the physics of damage mechanics and interactions. In this regard, based on Beaumont [1], "There is a need to link experience at levels between the macroscopic size with understanding at the micro-structural scale of the material." By discerning the reasons for damage initiation and its interactions with other forms of damage, materials can be refined and structural designs improved for further applications.

However, when it comes to making decisions related to predicting the remaining useful life (RUL), initiating shutdown procedures, and determining maintenance types, the necessity of a comprehensive health indicator (HI) becomes evident. In fact, HI serves as a macroscopic index. As stated in [1], "The macroscopic response of a composite material system and component reflects responses at all levels beneath." Thus, a HI should inherently encompass all other damage indexes (DI) pertaining to a structure, including those at microscopic and mesoscopic levels. However, despite advancements in understanding material deformation and fracture mechanisms through defect theory and crack mechanics, predicting failures in composite structures at a macroscopic level remains challenging [1]. Nevertheless, since the HI plays a pivotal role in *comprehensively* representing the structure's health status, its comprehensiveness can be defined and mutually measured by several criteria, which will be introduced in the subsequent chapters.

The HI serves as a valuable index, demonstrating the health level of an engineering system or structure [2, 3]. It acts as an intermediary, connecting the raw data collected through condition monitoring (CM) or structural health monitoring (SHM) techniques with the prognostic models used for RUL prediction [4]. One might question why we do not directly predict RUL from sensory data, bypassing the HI step. To answer this question, we must consider the definition of true RUL, which, logically and philosophically, follows a linearly decreasing pattern. It is natural to ponder the implications of uncertainties, such as accidents for humans or impact loading for structures, which could lead to a decline in the end-of-life (EoL) (see Figure 1.1(a)). However, in an ideal prognostic framework, these uncertainties should be anticipated and factored in advance, with the true RUL only definitively determined after the final failure and EoL. The same principle applies to maintenance and self-healing processes, which can extend the EoL. From another perspective, the relation between relative running time and relative RUL is fixed and linear. Tomorrow, in comparison to today, we will lose one day of our lives, regardless of whether we experience an accident or maintain a healthy lifestyle (see Figure 1.1(b)). It is crucial to emphasize that linking the consequences of these events to a decrease or increase in RUL represents a misinterpretation, as it misleadingly mixes the concept of RUL with health degradation (HI).

Consequently, uncertainties and progressive damage can only be continuously monitored and observed through the pattern of a HI, rather than a linear RUL graph. This establishes a tangible and interpretative connection between the HI and the engineering system, making it not only a suitable candidate for diagnostic purposes but also a vital in-

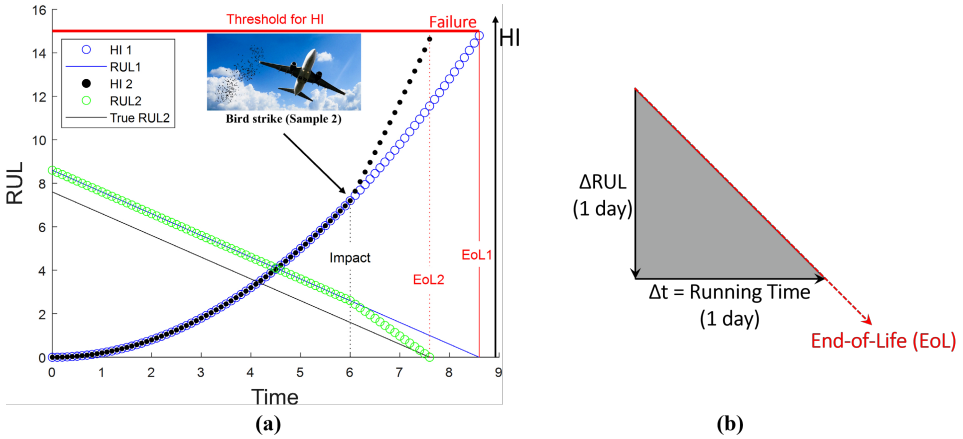


Figure 1.1: (a) RULs and HIs for two units with and without impact loading. (b) Relation between relative RUL and relative running time.

put for prognostic models used to predict RUL. In contrast, a RUL model primarily serves to determine the timing of maintenance rather than specifying the precise maintenance actions and decisions.

1.2 HIs for Diagnostics & Prognostics

A HI should be extracted from CM/SHM data for both diagnostic and prognostic purposes. Regarding diagnostics, a HI should be interpretable. This interpretability can be defined in two main aspects (see Figure 1.2): The first aspect pertains to the HI having a relation with the physical and mechanical behaviors of the engineering system or structure (referred to as 'unit' in this book). For example, attributes like stiffness of the structure, crack density in the structure, or different damage states during fatigue loading could be considered as attributes that a HI should possess. The second aspect involves understanding the relation between sensory monitoring data and the appropriate (based on prognostic criteria) HI, assuming we already know the connection between the mechanics of different damage types in the unit and monitoring data. In this context, as the monitoring data is interpretable thanks to its clear relationship with damage mechanisms, the HI becomes interpretable as well. This aspect can be further divided into two categories: I) interpreting the exact mathematical function between the HI and the monitoring data; and II) finding a basic correlation between the HI and the monitoring data. We refer to the first as an 'interpretable HI', which carries a deeper meaning than the second, which we label as an 'explainable HI'. With these definitions, an interpretable HI is inherently explainable, while an explainable HI may not necessarily be interpretable. In diagnostics, the author argues that an explainable HI is sufficient to link with the unit's physical behavior for better maintenance decisions.

Regarding prognostics, HIs designed for a group of similar units should meet established criteria as standard in the field of prognostics and health management (PHM), which will be elaborated in the next chapter. Briefly, the quality and suitability of HIs are usually

1

measured by three metrics: monotonicity (Mo), prognosability (Pr), and trendability (Tr) [5, 6], each addressing a specific property of the HI. All three are crucial for enhancing HI quality and increasing prognostic accuracy. However, extracting HIs that fulfill these criteria is very challenging, even when assuming informative SHM data is available, due to the inhomogeneous material characteristics, variable operational conditions, stochastic activation and interaction of damage mechanisms, and uncertainties. These criteria are formulated based on degradation physics and EoL fact.

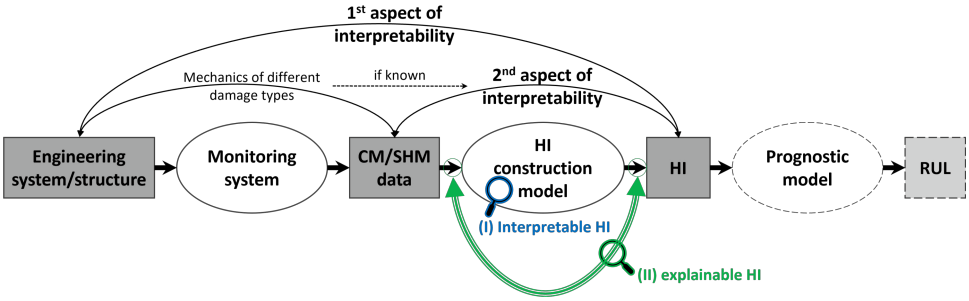


Figure 1.2: Different aspects of the interpretability of a HI.

1.3 Which SHM Technique: Fusion of Multiple Techniques?

While various SHM techniques are available for structural monitoring, the crucial question is which technique can provide the most informative and reliable data for creating HIs. Is a single SHM technique sufficient, or should multiple techniques be fused together?

Prognostic and HI construction models commonly exhibit historical dependence (HD). Considering the correlation between historical SHM data, from the healthy state to the present, is essential to enhance HI and RUL prediction models [7–9]. This consideration is particularly critical for passive SHM techniques like acoustic emission (AE), which captures temporary signals related to damage initiation and propagation, lacking explicit information about damage size, structural stiffness, or other historical-dependent factors. Passive techniques inherently provide insights into how and when damages occur, but the historical data requires processing to extract HD characteristics, such as structural stiffness [10].

In contrast, active SHM techniques, like guided waves (GW), can estimate damage size, location, and stiffness reduction in the structure from *the most recent data*[1], covering the progression of damages from the beginning up to the present [11, 12]. However, active techniques do not capture information about damage initiation and propagation, including propagation rates, as precisely as passive techniques [13]. Thus, there is a trade-off between passive and active SHM techniques.

Incorporating HD information into a model for passive SHM techniques could be an effective solution, providing a comprehensive understanding of damage initiation and prop-

[1]Historical data is not required; possibly just one pristine inspection is required as the baseline (see Chapter 6).

agation. Nevertheless, both RUL prediction and HI construction models have a common drawback - they are historical-dependent. This necessitates the consideration of the temporal relationship between historical data from the starting point to the present moment to enhance their performance [6, 14, 15]. However, a lack of prior information, whether due to operational issues or unavailable monitoring systems, can hamper these models' efficiency. In this context, a robust SHM technique is required to extract informative historical-independent evidence, while a suitable model is needed to create HIs from this data.

In conclusion, the question remains: which SHM technique can provide the most informative yet reliable data for building HIs? Beyond data- and feature-level fusion for each specific technique, should different SHM techniques be fused? Can the fusion of a passive SHM technique (e.g., AE), providing insights into damage initiation and propagation, and an active SHM technique (e.g., GW), offering information about final damage characteristics, potentially lead to more robust and reliable HIs?

1.4 Research Goal & Scope

Primary, secondary, and other objectives of this thesis are as follows:

Primary objectives: The primary objective of this research is to develop an innovative framework for designing comprehensive HIs that meet prognostic requirements. This framework leverages raw SHM data and places a specific focus on the fatigue life of aerospace composite structures.

To enhance the accuracy of prognostic criteria in data-driven frameworks, this research will refine the metrics. Instead of considering all HIs generated from both training and testing phases, the revised metrics will primarily consider testing units. This adjustment aims to prevent the undue influence of training units on final scores, ensuring a more robust evaluation of the test unit's performance and its deviation from the training units.

The framework itself consists of several key stages: data acquisition, pre-processing (PP), signal processing (SP), feature extraction (FE), and feature fusion (FF). Feature fusion, in particular, plays a pivotal role in implementing the prognostic criteria. To achieve this, a novel semi-supervised learning paradigm is introduced for fusing features, with the goal of achieving the quality necessary for prognostics. This developed paradigm has broad applications, extending beyond composite structures to areas like commercial turbofan engines.

Two SHM techniques, AE as a passive technique and GW as an active technique, are employed to monitor composite structures, aiming to answer the central question raised earlier: "which SHM technique? Fusion of multiple techniques?" For each technique, specific algorithms for PP, SP, FE, and FF are developed and put into practice. The FE and FF processes primarily draw from elements of artificial intelligence (AI) and machine learning (ML). The experimental composite specimens monitored by these techniques encounter various uncertainties and unexpected phenomena during their operational service life.

Secondary objective: Furthermore, this research involves the development of a historical-independent HI construction model based on the GW technique. This model is tailored for aerospace composite structures under fatigue loading conditions, offering

flexibility to accommodate different GW sensor configurations, networks, and setups to address a variety of layups, geometries, and loading conditions.

Other objectives: In pursuit of interpretability, this research encompasses two initiatives:

1. The development of HIs that show incremental steps, potentially corresponding to distinct damage states within the context of composite structures, where these incremental steps can provide valuable insights for informing future state-based RUL prediction models
2. The construction of interpretable HIs through the development of an interpretable neural network designed for commercial turbofan engines

Expectations: It is anticipated that the outcomes of this thesis will contribute to advancing the field of PHM, with a specific focus on aerospace composite structures that face challenges due to the stochastic nature of damage accumulation and the occurrence of uncertain events, such as impacts, during operational conditions.

1.5 Thesis Outline

This thesis is composed of 8 chapters and appendices, as follows:

- **Chapter 2: Literature Review**

  Chapter 2 provides a review of the existing literature in the field of PHM. It delves into the concepts of physical and virtual HIs and explores how prognostic criteria have been utilized in prior works. The chapter particularly focuses on their implementation in the context of composite structures. Moreover, it opens an argument about interpretability vs. explainability vs. accuracy in this context.

- **Chapter 3: Methodology**

  In Chapter 3, prognostic criteria are first introduced for qualifying HIs, referred to as HIs' criteria. A refined version of these criteria is presented, considering only test units. Then, the core methodology is outlined, featuring the introduction of a new semi-supervised learning paradigm for SHM data fusion toward HI construction. In the end, the chapter explores interpretability by drawing comparisons between composite structures and commercial turbofan engines. Following these discussions, it introduces a modified multiplicative neuron with the specific purpose of constructing explainable HIs, primarily tailored for use in the domain of commercial turbofan engines.

- **Chapter 4: Data Analysis and Machine Learning**

  Chapter 4 provides essential background information on data analysis, processing, and mining. It offers brief explanations of various data analysis operators, functions, and algorithms that will be utilized in this thesis. The primary components of this chapter include pre-processing (PP), signal processing (SP), dimension reduction with a focus on principal component analysis (PCA)-based algorithms, statistical features, and deep learning.

- **Chapter 5: Designing HI for T-Stiffener Composite Panels using Acoustic Emission**

  Chapter 5 demonstrates the proposed methodology for monitoring twelve single T-stiffener carbon fiber-reinforced polymer (CFRP) specimens under compression-compression fatigue loading using the AE technique. Three distinct frameworks incorporating various pre-processing, signal processing, feature extraction, and feature fusion strategies are developed to generate qualified HIs. The chapter explores the motivations behind designing these frameworks and presents the results of ablation experiments, dataset divisions, and leave-one-out cross-validation to assess the approaches' generalizability.

- **Chapter 6: Designing HI for T-Stiffener Composite Panels and Dogbone Specimens using Guided Waves**

  The methodology for the GW monitoring technique is demonstrated in Chapter 6, involving the monitoring of five single T-stiffener CFRP panels under compression-fatigue loading as well as eleven dogbone CFRP specimens with three different layups subjected to tension-fatigue loading. This chapter's motivation goes beyond answering the question, "which SHM technique: Fusion of multiple techniques?" to include the design of qualified historical-independent HIs for aerospace composite structures. To make sure the method was generalizable, leave-one-out cross-validation was also taken into consideration.

- **Chapter 7: Fusion of Acoustic Emission and Guided Waves Techniques**

  Chapter 7 explores the fusion of AE and GW techniques. It combines the HI construction models developed for single T-stiffener CFRP panels using each technique. The chapter aims to answer the question, "which SHM technique: Fusion of multiple techniques?" by comparing HIs generated from AE, LW, and the fusion of both techniques.

- **Chapter 8: Conclusions and Recommendations**

  The thesis's findings are outlined in Chapter 9, along with conclusions and suggestions for future research.

- **Appendices: Interpretable HIs for Commercial Turbofan Engines**

  Appendix A.1 presents the results of an interpretable neural network that was developed to construct interpretable HIs for commercial turbofan engines. It includes newly modified multiplicative layers with discretized weights.

References

[1] P. W. Beaumont, "The structural integrity of composite materials and long-life implementation of composite structures," *Applied Composite Materials*, vol. 27, no. 5, pp. 449–478, 2020.

[2] H. Qiu, J. Lee, J. Lin, and G. Yu, "Robust performance degradation assessment methods for enhanced rolling element bearing prognostics," *Advanced Engineering Informatics*, vol. 17, no. 3-4, pp. 127–140, 2003.

[3] C. Song, K. Liu, and X. Zhang, "Integration of data-level fusion model and kernel methods for degradation modeling and prognostic analysis," *IEEE Transactions on Reliability*, vol. 67, no. 2, pp. 640–650, 2017.

[4] L. Guo, N. Li, F. Jia, Y. Lei, and J. Lin, "A recurrent neural network based health indicator for remaining useful life prediction of bearings," *Neurocomputing*, vol. 240, pp. 98–109, 2017.

[5] J. B. Coble, "Merging data sources to predict remaining useful life–an automated method to identify prognostic parameters," 2010.

[6] M. Moradi, J. Chiachío, and D. Zarouchas, "Developing health indicators for composite structures based on a two-stage semi-supervised machine learning model using acoustic emission data," in *Proceedings of the 10th ECCOMAS Thematic Conference on Smart Structures and Materials*, vol. 10, 2023.

[7] M. Ahmadivala, *Towards optimal maintenance planning of existing structures based on time-dependent reliability analysis.* PhD thesis, Université Clermont Auvergne [2017-2020], 2020.

[8] A. J. Valkonen, *Exploring Structural Health Monitoring Value of Information Based on Remaining Useful Life Extension Potential.* PhD thesis, Princeton University, 2023.

[9] P. Kamranfar, *Machine Learning Enabled Health Monitoring And Diagnosis of Engineering Systems.* PhD thesis, George Mason University, 2023.

[10] M. Saeedifar and D. Zarouchas, "Damage characterization of laminated composites using acoustic emission: A review," *Composites Part B: Engineering*, vol. 195, p. 108039, 2020.

[11] B. Zhang, D. Yang, X. Hong, and G. Jin, "Deep emulational semi-supervised knowledge probability imaging method for plate structural health monitoring using guided waves," *Engineering with Computers*, vol. 38, no. 5, pp. 4151–4166, 2022.

[12] L. Tang, Y. Li, Q. Bao, W. Hu, Q. Wang, Z. Su, and D. Yue, "Quantitative identification of damage in composite structures using sparse sensor arrays and multi-domain-feature fusion of guided waves," *Measurement*, vol. 208, p. 112482, 2023.

[13] A. A. Ijjeh and P. Kudela, "Deep learning based segmentation using full wavefield processing for delamination identification: A comparative study," *Mechanical Systems and Signal Processing*, vol. 168, p. 108671, 2022.

[14] T. Peng, Y. Liu, A. Saxena, and K. Goebel, "In-situ fatigue life prognosis for composite laminates based on stiffness degradation," *Composite Structures*, vol. 132, pp. 155–165, 2015.

[15] M. Corbetta, C. Sbarufatti, M. Giglio, A. Saxena, and K. Goebel, "A bayesian framework for fatigue life prediction of composite laminates under co-existing matrix cracks and delamination," *Composite Structures*, vol. 187, pp. 58–70, 2018.

2

Literature Review

Within the chapter, damage, SHM, and PHM in composite structures; physical and virtual HIs; prognostic criteria implementation in HI design; and interpretability and explainability trade-offs with accuracy are covered.

2.1 Introduction

In this chapter, a review of composite structures and their associated damage is presented (Section 2.2). Following this, the applications of SHM techniques in the realm of composite structures are outlined in Section 2.3. The discussion then extends to the field of PHM, with a specific emphasis on its application to composite structures. Subsequently, in Section 2.4, HIs are introduced and categorized into two main types: physical and virtual. The main part of this chapter (subsection 2.4.3) explores how prognostic criteria are integrated into the design process of virtual HIs. Lastly, the chapter concludes with Section 2.5, which opens a discussion on the differences and trade-offs of interpretability versus explainability versus accuracy within the context of HI design.

2.2 Failure Mechanisms in Composite Structures

Composite structures play a crucial role in industries such as aerospace, wind energy, and naval applications, serving as primary materials in high-performance structures. Their increasing use is driven by unique mechanical performance, reduced weight, and high durability, resulting in enhanced safety and cost savings, primarily through reduced fuel consumption. However, composite structures are vulnerable to various forms of damage during operation, leading to a diverse range of structural failure scenarios. Unlike conventional structures, they may experience a broader spectrum of failure modes, including tensile, compressive, or shear fracture of the matrix, bond failure of the fiber-matrix interface, and tensile or compressive (buckling) failure of the fibers.

Categorized by the size of the damage and the affected region (fiber or matrix), Figure 2.1 illustrates different types of defects in composite structures [1, 2]. Among the most common and critical damage types are those resulting from impact and cyclic fatigue loading. The severity of damage, depending on its type and size, can significantly reduce a composite structure's residual strength [3], manifesting as defects in the fiber, matrix, or their interface [4].

Assessing damage severity is crucial for determining a composite component's remaining service life and influencing the structure's overall damage tolerance. While certain

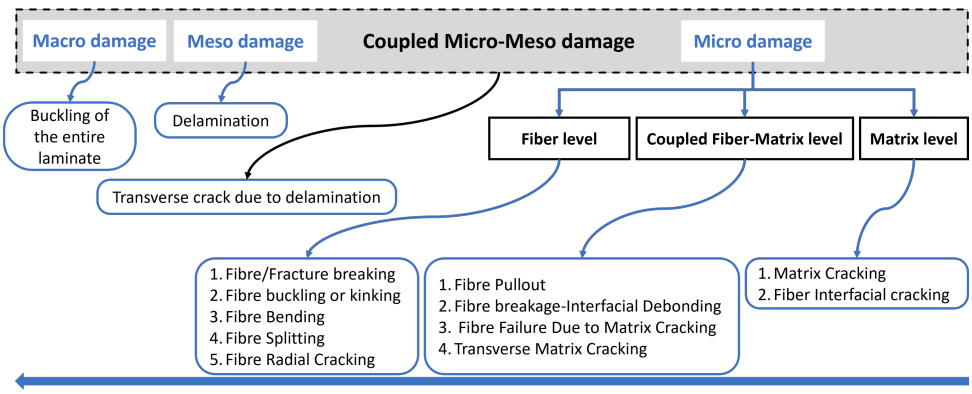


Figure 2.1: Types of damage in composite structures (adopted from [2] and modified based on [1]).

types of damage may have minimal immediate effects on residual strength, their impact may worsen over time, especially when compounded by operational and environmental influences [5, 6]. This highlights the importance of frequent and thorough monitoring (SHM) to address the complexities associated with the multivariate damage process in composite structures.

2.3 SHM for Composite Structures

According to the SAE Standard ARP646, the definition of SHM is *"the process of acquiring and analyzing data from on-board sensors to evaluate the health of a structure"*, which is a consensus from the Aerospace Industry Steering Committee on SHM. Representatives of the essential industries and universities active in the field of SHM composed it. Three main principles of SHM systems are [7]:

1. To perform an automated inspection, several inspection systems permanently monitor the structure/system. The inspection systems can be a network of sensors (such as GW, AE, fiber Bragg grating (FBG), distributed fiber optic sensing (DFOS), etc.) attached to the structure. This perspective is the basic difference from non-destructive testing (NDT) procedures.

2. Since in SHM, a large amount of data has to be analyzed in real-time, on-board computation and data handling facilities are needed.

3. To estimate the damage index (DI) and gain information about the existence, locations, types, propagation speed, and other attributes of damages, the current data is compared with the previously stored data from the pristine structure.

The field of SHM plays a crucial role in diagnosing damage within various composite structures. NDT is another field with a big overlap with SHM, which encompasses a range of methods for identifying damage without causing harm to the structure being examined. The reliability of NDT methods such as Ultrasonic, X-ray, and infrared thermography (IRT) has been firmly proven. These techniques are particularly valuable for their cost and time-saving benefits in system evaluation. While SHM techniques are often assumed as techniques with sensors attached to the structure, the author suggests that any continuous monitoring technique qualifies as an SHM method, regardless of physical contact with the structure. The primary distinguishing factor between SHM and NDT lies in the inspection frequency. For instance, considering the inspection of an aircraft using GW exclusively during an unloading condition (post-landing), the inspection frequency aligns with that of using IRT via a drone in the same condition (post-landing). In this context, if GW is categorized as an SHM technique, then IRT falls within the same classification. Another scenario involves structures on the ground, such as wind turbines, which can be continually monitored using optical techniques like IRT [8], even without attached sensors to the structure. Thus, the final purpose of SHM, which is a continues monitoring, itself defines a technique as an SHM or NDT method.

Despite differences between NDT and SHM, both can be considered under the umbrella term non-destructive evaluation (NDE) [7], which their complementary information in this regard can be studied in Ref. [9] and Ref. [10] as new surveys (from 2000 to

2017) and (from 2000 to 2023), respectively. The 'Handbook of Advanced Nondestructive Evaluation' [11] is strongly suggested as a unique outline of advanced NDE techniques. Non-destructive testing and evaluation (NDTE) techniques are listed in Figure 2.2, along with a list of subcategories [2]. As demonstrated in Figure 2.3, each of these methods can be used to address a particular spectrum of damage in composite structures. One major drawback of these methods is that it is hard to perform the evaluation process without prior knowledge of the general location of the damage.

Advantages and disadvantages of different SHM techniques, including AE, GW, ultrasound, strain, vibration, eddy current testing, IRT, radioscopy/radiography testing, and visual inspection, can be found in different sources, like Ref. [12] with a focus on diagnosis and prognosis of wind turbines and Ref. [13] with a focus on diagnosis of wind turbine

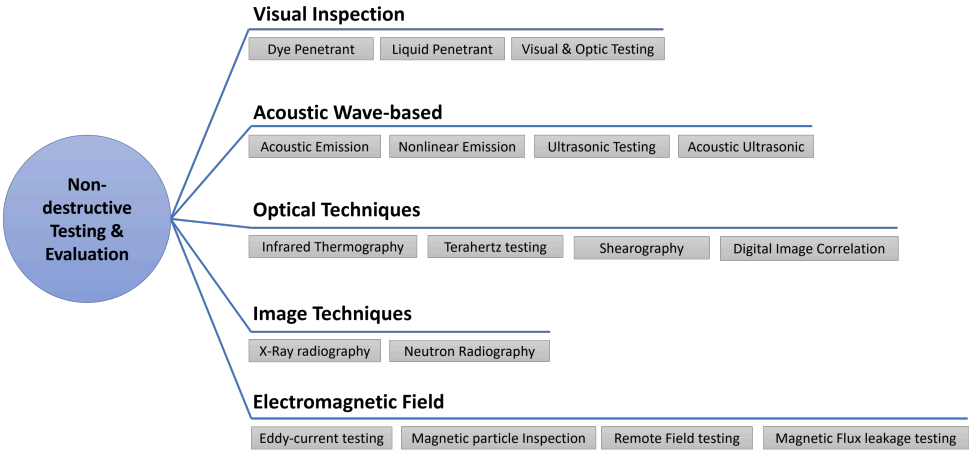


Figure 2.2: Classifications of various non-destructive testing and evaluation techniques (NDTE) (adopted from [2]).

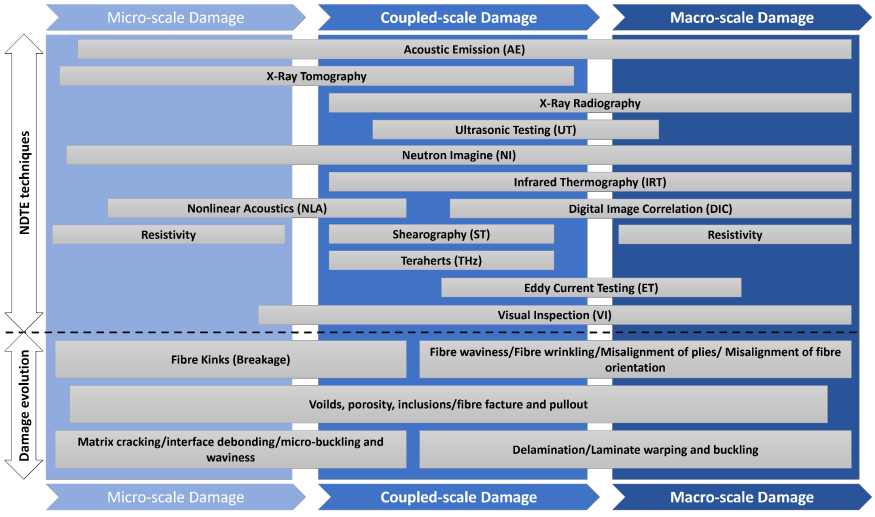


Figure 2.3: The spectrum of damage to which various NDTE methods can be used (adopted from [2]).

blades, which are mainly composite structures.

Among SHM methods, AE stands out as a popular technique for composite structures. It involves capturing signals generated by various damage mechanisms, providing insights into the progression of damage during the material's lifetime [14, 15]. AE can distinguish different damage mechanisms by studying various features, such as peak frequency, amplitude, duration, and energy [16, 17]. However, challenges include susceptibility to external noise, requiring careful parameter tuning and efficient post-processing [18, 19].

GWs are widely utilized in the aviation industry for thin-walled composite structures. This method allows the interrogation of the entire structure with a low attenuation ratio, detecting small-sized damages with high accuracy [20, 21]. Challenges include interference from reflection and scattering phenomena due to structural boundaries, as well as sensitivity to environmental conditions [22, 23].

Both AE and GW fall under the main category of acoustic wave-based methods [10], which involve identifying and monitoring sonic and ultrasonic stress waves. Stress waves can be actively imparted (as in GW) or passively emitted (as in AE). In order to identify internal defects and characterize materials, they are usually carried out within the elastic material range. While both AE and GW originate from the same physics, they measure different characteristics of the structure. AE, a passive method, is capable of measuring the initiation and propagation of damage, whereas GW, an active method, assesses the state of damage (e.g., size). Although AE can be used to determine the location and size of damage and GW may be employed to approximate the speed of propagation or initiation of damage, it is essential to consider historical dependence (HD) or time dependence in both SHM data from prior evaluations up to the current state to do so. This consideration, along with the relevant advanced post-processing, is imperative for these purposes. Thus, here, the assessment of each SHM method's capability was meant based on their original data, regardless of post-processing algorithms afterward.

2.3.1 Acoustic emission (AE)

As mentioned in the previous chapter, a common characteristic of prognostic and HI construction models is their HD. It is crucial to consider the correlation between historical SHM data, starting from a healthy state to the current time, in order to enhance the performance of HI and RUL prediction models [24–26]. This issue is particularly essential for passive SHM methods, such as AE, which capture temporary signals resulting from damage initiation and propagation rather than specific information about the size of the damage, structural stiffness, or other HD factors. Thus, a valuable solution could be incorporating HD into a model for passive SHM methods, allowing a more comprehensive understanding of damage initiation and propagation, making it particularly suitable for scenarios where such historical data is crucial for prognostic applications (refer to models presented in Chapter 5).

2.3.2 Guided waves (GW)

Although considering HD into a model for passive SHM methods could be a solution, the fact that RUL prediction and HI construction models are historical-dependent is a common drawback. In other words, prognostic and HI construction models function less efficiently when prior information, either entirely or partially from the beginning, is miss-

ing. This might occur for various reasons during operation due to unavailable monitoring systems, misfunctioned sensors, etc. In this regard, a robust SHM method is needed to extract informative, historical-independent patterns. The GW technique can be a potential candidate that can generate such historical-independent patterns [27]. However, GWs pose a challenge as they not only convey information about damage but also carry signals that are susceptible to interference from reflection and scattering phenomena at structural boundaries. Environmental conditions can also further affect the reliability of GW signals [28, 29]. Moreover, GWs are dispersive and have a multimodal nature, meaning that many wave modes may exist in a recorded signal, and their characterization depends on the frequency, material stiffness, density, and thickness of the structure [30]. Therefore, the diagnosis and prognosis of composite structures via GWs is a hard task when it comes to correlating the GW signals with the health state of the entire structure. In addition to the previously discussed aspects, it is important to note that translating GW data to the appropriate HI value at each time step when GW inspections are conducted is a challenging task. In fact, a model is needed to map thousands of data points (as can be seen in experimental campaigns that generated extensive datasets—cite NASA and ReMAP in Chapter 6) to a single HI value at the current time, regardless of the prior HIs. To address this challenge, data-driven approaches, especially AI, have drawn attention in diagnostic [31–34] and prognostic [35, 36] applications thanks to their ability to discover complex and nonlinear relationships between data (refer to the model presented in Chapter 6).

2.3.3 Fusion of AE and GW

As mentioned earlier, there is a trade-off between passive and active SHM methods. The former (e.g., AE) provides details about damage growth but requires historical data to be post-processed for extracting HD characteristics, such as defect dimensions. Meanwhile, the latter (e.g., GW) can estimate the final damage state but lacks information about damage growth. While both AE and GW can be utilized with advanced data analysis algorithms to offer both temporal (damage initiation and growth) and spatial (damage state, location, and size) information, they cannot surpass each other's optimum performance. Despite applying the best ideal processing algorithms for each passive or active SHM technique, AE can achieve optimal performance regarding temporal information, and GW can achieve optimal performance regarding spatial information. This limitation arises from the fact that each type of SHM data contains limited useful information, and data analysis algorithms cannot perform miracles. Figure 2.4 illustrates this concept, assuming that post-processing algorithms and fusion models have the best performances and function perfectly. Therefore, the fusion of SHM techniques could be beneficial for optimally exploiting temporal and spatial information, preventing the loss of valuable information.

In this subsection, prevalent information fusion techniques are briefly reviewed, and the classes and levels that will be used in this work are hinted at. The information fusion strategies can be conceptualized based on factors like sensor relations, input-output relations, and decision relations, as illustrated in Figure 2.5. Figure 2.6 outlines various classifications of data fusion, encompassing Whyte's, Dasarathy's, joint directors of laboratories (JDL), architecture, and abstraction classifications, considering different processing levels. A comprehensive literature review on each data fusion category, focusing on types and applications in SHM systems, is available in Ref. [10].

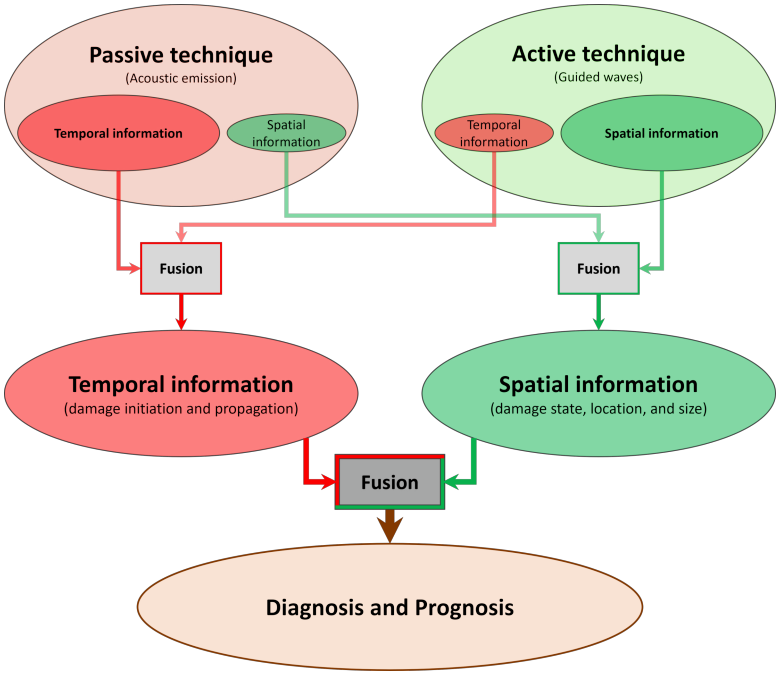


Figure 2.4: Conceptual representation of temporal and spatial information exploitation from passive and active SHM techniques and their fusion toward diagnosis and prognosis.

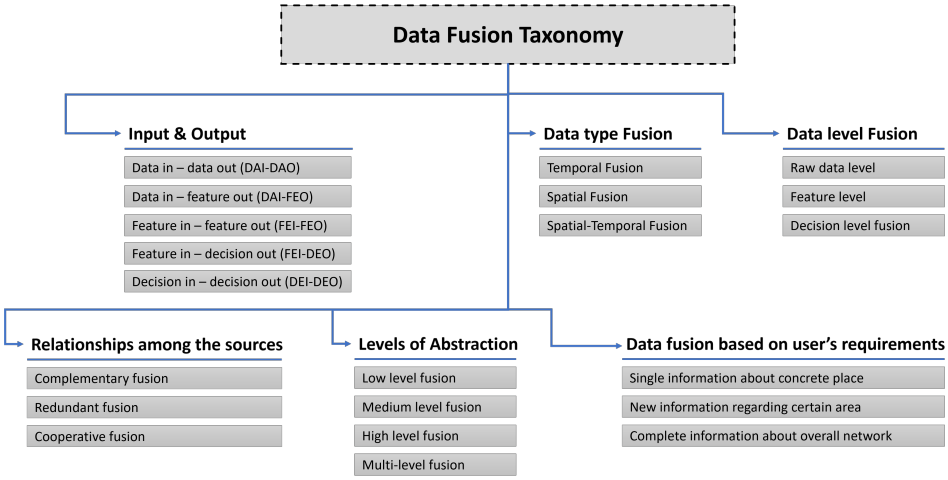


Figure 2.5: Information fusion classifications based on different factors (adopted from [10]).

According to Figure 2.5, the present work will conduct fusion in the following classes:

- Based on input-output: fusion in classes of DAI-DAO, DAI-FEO, and FEI-FEO, considering that HI is a high-level feature
- Based on relationships among sources: complementary (among sensors of AE

and/or GW), redundant (among sensors of AE and/or GW), and cooperative (besides among sensors of each technique, between AE and GW techniques) fusion

- Based on levels of abstraction: low-level (raw data level fusion), medium-level (feature level fusion), and multi-level fusion
- Based on data type fusion: temporal, spatial, and spatial-temporal fusion, as mentioned earlier
- Based on data level fusion: raw data level and feature level
- Based on the user's requirements: depending on how the area and sensory network on the composite specimen would be considered

According to Figure 2.6, the present work will conduct fusion in the following classes:

- Based on Whyte's classification: classes of Dt_{In}-Dt_{Out}, Dt_{In}-Ft_{Out}, and Ft_{In}-Ft_{Out}
- Based on Dasarathy's classification: complementary, redundant, and cooperative fusion, similar to Figure 2.5
- Based on JDL classification: source pre-processing (L0) and object refinement (L1)
- Based on architecture classification: distributed architecture
- Based on abstraction classification: the same as in Figure 2.5, at both single-level and multi-level fusion

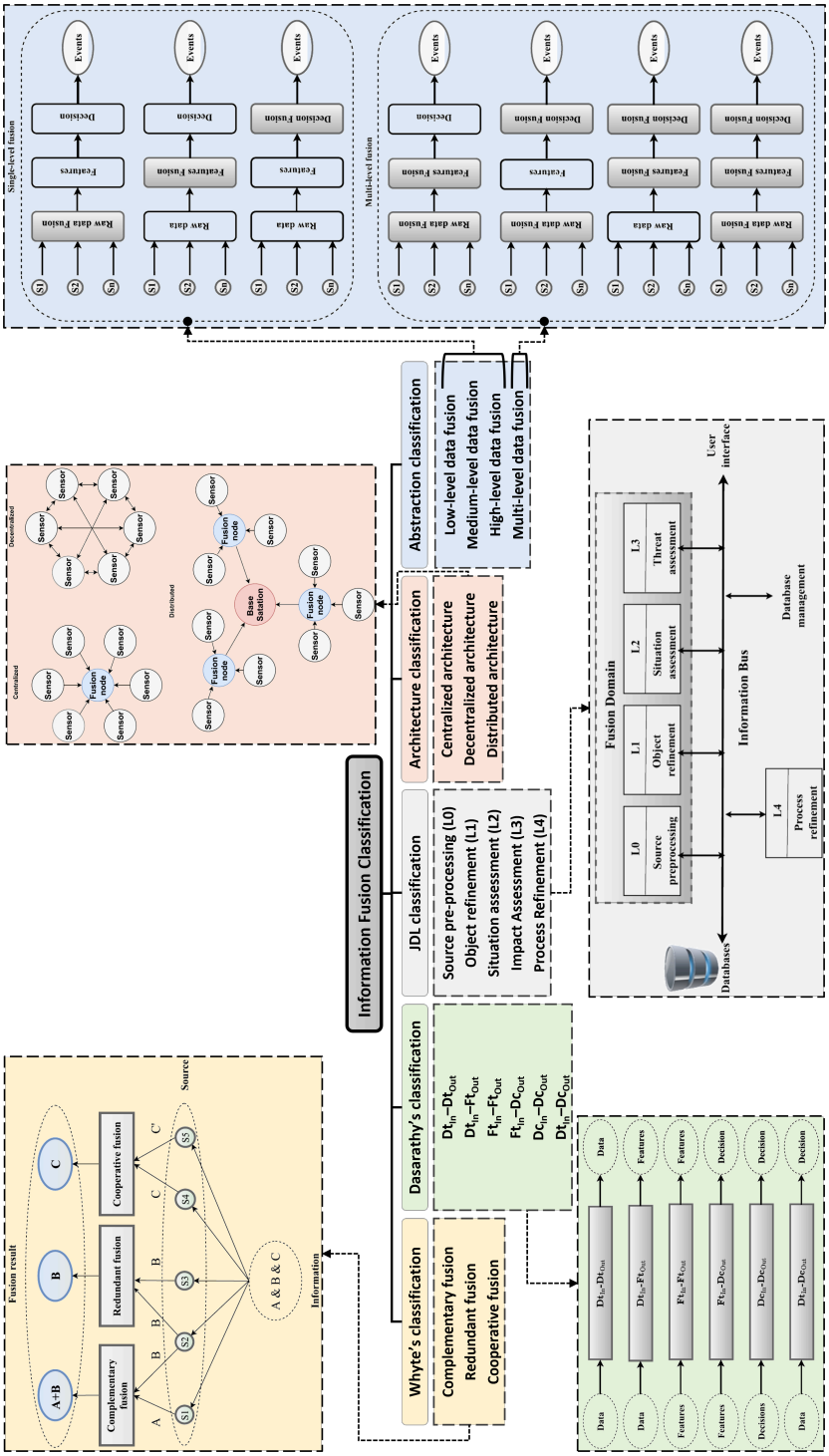


Figure 2.6: Classifications of information fusion (adopted from [10]).

2.4 PHM for Composite Structures

SHM plays a pivotal role in the prognostics and health management (PHM) framework, aiming to propel the industry toward condition-based maintenance (CBM) policies, thereby boosting availability and curbing operational downtime and costs [37, 38]. The transition to a CBM paradigm within a PHM framework necessitates reliable and robust SHM methodologies to facilitate informed decision-making.

PHM naturally extends from SHM by updating predictions of RUL in a timely manner using sensor data. For accurate RUL predictions, a HI suitable for incorporation into a prognostic model is essential [39, 40]. HI, a distinctive feature derived from SHM data, reflects the health (or damage) status of the monitored structure or system [41]. When the HI surpasses a predefined threshold, the SHM system triggers alarms, prompting actions like system shutdown, maintenance, or replacement. Consequently, HI serves as a crucial link and prominent feature between raw signals and the prognostic model, directly influencing RUL predictions.

Unlike RUL, which is often assumed to follow a linear degradation model [42], HIs are nonlinear due to the inherent nonlinear nature of damage propagation and accumulation [36, 43, 44]. This nonlinearity proves valuable for analyzing and connecting with the mechanical behavior of the structure. However, it is important to note that HIs are not exclusively sensitive to damage; they may also respond to abnormalities in environmental conditions and operations.

2.5 Health Indicators

In order to visualize data and continuously characterize the health state of the structure over its entire life, HIs can be informative and helpful [45]. The available data-driven RUL prognostic methods, such as artificial intelligence and statistical-based models, can also be effectively integrated with the fusion-based HIs [46]. Depending upon whether a HI has any physical senses, system performance data is typically divided into two categories: the physical health indicator (PHI) and the virtual health indicator (VHI) [47]. Before delving into the development of a framework for designing a HI, it is crucial to outline the expectations—constraints derived from the physics of the problem—associated with HIs. Subsequently, an exploration is undertaken to incorporate these constraints effectively within the design framework.

2.5.1 Evaluation criteria (constraints)

If no maintenance and self-healing take place, a structure's HI should decrease throughout operational conditions due to damage accumulation. This fact should be incorporated into the design of a HI and examined using a metric known as monotonicity (Mo) [48, 49]. The comprehensive HIs of an ensemble of associated units (engineering systems or structures) that have reached their end-of-life (EoL) should ideally arrive at the same value, signifying the failure threshold. However, HIs at the EoL change and do not always end up with an identical value; this discrepancy can be quantified using a metric called prognosability (Pr) [49]. Moreover, if the HIs demonstrate a consistent correlation with usage time, they are deemed more predictable. The trendability (Tr) criterion gauges the resemblance between the trends of HIs [49–51]. While the first two assessment criteria (Mo and Pr) are consid-

ered factual, achieving maximum Tr may be challenging due to stochastic events and the uncertain nature of influential factors. Nevertheless, striving for HIs with high Tr remains a goal to improve the accuracy of RUL predictions. In the context of prognostics, which is the main focus of this work, a HI should fulfill these evaluation/prognostic criteria: Mo, Pr, and Tr [52, 53].

2.5.2 Physical health indicators (PHI)

Since health management has been given more attention and begun earlier for some particular industrial components, such as batteries and rotating machinery equipment, PHIs are relatively well known and helpful. In other words, the long-term and vital use of such components, as well as their history in the industry, in turn results in providing more datasets as well as more solid physical, analytical backgrounds [36]. For instance, according to physical knowledge about batteries, the capacity data of lithium-ion batteries can be considered as a proper HI [54]. It was recently said that, ‘‘To the best of our knowledge, the capacity is a typical performance indicator to monitor the health status of the battery and determine whether the battery requires replacement; thus, we adopt such indicator for the RUL prediction’’ [55]. For rolling element bearings, Ref. [56] utilized root mean square (RMS) as ‘‘a simple and practical HI, which is widely used in bearings residual life prediction’’. Also, relative RMS (RRMS) could be more robust to characterize the degradation process of bearings [57]. However, to the best of the author's knowledge, no certain and promising PHI has yet been developed for composite structures, especially for structures subjected to cyclic fatigue loading, which is one of the most critical conditions for composite structures. This can be due to complex scenarios of progressive damage and different types of damage, which in turn are dependent on the type of fibers, matrix, fabrication, curing, boundary and environmental conditions, loading, etc.

Despite the fact that various articles have introduced PHIs (axial strain) utilizing DIC data [58], or the size or number of cracks have been considered as PHIs [59], they are not appropriate for genuine SHM and implementations. Concerning the first subject, methods like DIC are not yet considered SHM because of their numerous drawbacks, including the necessity of painting, particular illumination requirements, problematic calibration, etc. The DIC method is also limited to measuring the strain or deformation at the surface of the structure in the field of view of the camera, making it more effective for plates due to the lack of access to the rear of the surface. Such methods are usually used to validate other aspects of research, such as finite element modeling [60] or other NDT/SHM techniques [61]. For the latter subject, it may be possible to correctly monitor and take into account the number or size of fractures as a HI in isotropic materials such as aluminum [59], but this is not easily practical for composite materials and complicated structures. In reality, a variety of impactful micro- and macro-damages will emerge, such as cracks and delaminations, which not only emerge randomly throughout the structure [62] but are also concealed in various quantities among the various layers of the composite material [63].

2.5.3 Virtual health indicators (VHI)

In contrast to PHI, VHI cannot be easily interpreted and realized in order to make an understandable connection to the physical implications. However, VHI can be designed and

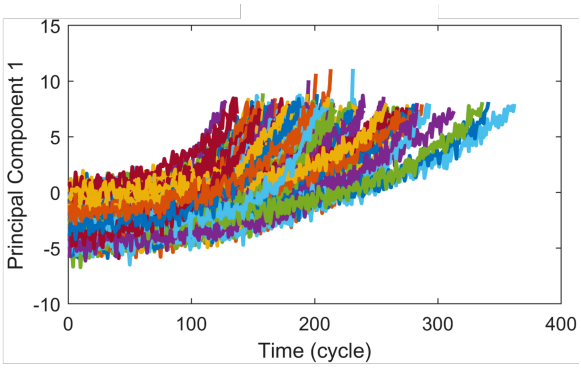
optimized based on the intended purposes, such as prognostics. For example, if a monotonic, trendable, and prognosable behavior can be embedded into an objective function [64] which is supposed to be used within a data-driven model, the resulted VHI is suited to the next step, the prognostic model.

Linear-based feature extraction and selection methods are suitable enough to provide an acceptable HI for some applications. For instance, principal component analysis (PCA) applied to gear vibration signals was able to extract a proper HI [65]. This method was also used to successfully construct the HI for the CMAPSS dataset (the turbofan engine degradation dataset), the PHM08 dataset (Prognostics Data Challenge Dataset), and the N-CMAPSS dataset (the new CMAPSS dataset) [66]. However, the PCA method does not generate suitable HIs for the ReMAP[1] dataset (single-stiffener composite panels under compression-compression fatigue loading studied in this work). Figure 2.7 demonstrates the 1^{st} principal component (PC) obtained by the PCA model on the CMAPSS dataset in comparison with the ReMAP dataset. It is clear that, in accordance with monotonicity, trendability, and prognosability (the three main intended evaluation criteria in PHM and this work), this method is ineffective for building HI utilizing either AE low-level features (amplitude, rise time, duration, energy, counts, and RMS) or AE high-level features (the features that have been extracted after signal processing and will be discussed in the current study).

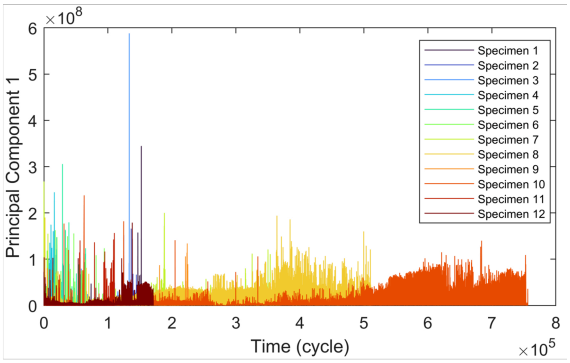
PCA, which is built on a linear reform of the original data, reveals its limitations whenever faced with inhomogeneity and time-varying correlations of component degradations. To address this shortcoming in the case of nonlinear data, a number of PCA modifications have been developed, including kernel PCA (KPCA), greedy KPCA (GKPCA)[67], PCA-based K-nearest neighbors, and PCA-based Gaussian mixture models [68]. However, the computational cost of these methods can be high, particularly when dealing with large datasets [69, 70]. For instance, KPCA necessitates computing the kernel matrix for every pair of input features given a single kernel, assuming that the right kernel has been selected in theory. Also, the potential degradation characteristics may not be effectively mined using the majority of the above conventional techniques, which are only applicable to datasets with a clear degradation trend [71].

In applications with massive amounts of data, artificial neural networks (ANNs) and deep learning (DL) algorithms can be utilized to build HIs without requiring a lot of domain knowledge. On the other hand, DL is a promising solution if enough labeled data is available. For some cases, like the current work, not only is enough data unavailable, but they are also unlabeled. In this regard, the limited available data should be efficiently processed and fit to the logical pattern according to the physics of the problem under evaluation. Thus, by taking advantage of FE on the input side (which leads to providing informative and labelable inputs) and inducing the intended desirable behavior, inspired by the physics of HI and its relation with the RUL of a structure, on the output side of the model (which leads to simulating labels), we try to approach the problem in this work.

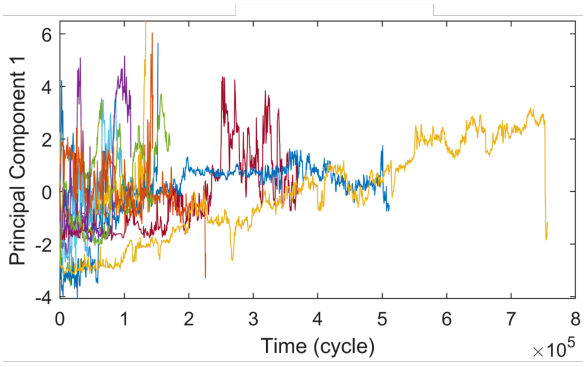
[1]ReMAP: Real-time Condition-based Maintenance for Adaptive Aircraft Maintenance Planning. https://h2020-remap.eu/.



(a) CMAPSS



(b) ReMAP: AE low-level features



(c) ReMAP: AE high-level features

Figure 2.7: First principal component calculated using PCA on (a) raw data in the CMAPSS dataset, (b) acoustic emission low-level features in the ReMAP dataset, and (c) acoustic emission high-level features the in ReMAP dataset [36].

Common & traditional procedures to extract HIs

The common procedure to extract a HI that can be employed to predict the RUL of a unit is to select the best features (after FE) in accordance with prognostic criteria as the HI

or the main constitutive components of the HI [72]. In this scenario, some features will be overlooked since they do not meet the criteria, while they might be useful since their fusion may comply with the intended specifications. To overcome this shortcoming, the prognostic criteria can play a supervising role in the construction process of HI rather than being only a measurement tool of HI's quality. For example, a predefined function with a set of polynomial components [64] can be considered to fuse features, in which the coefficients of the polynomial components are unknown and have to be determined. In this regard, combining the prognostic criteria into a ''fitness'' function could be considered as an objective function for an optimization problem [73]. In this approach, although a polynomial series might construct a proper HI function, the other components, e.g., logarithmic and exponential ones, might construct a function with more monotonic, trendable, and prognosable behavior. In fact, the fusion function is limited to only polynomial kernels in this approach, whereas other mathematical kernels and combinations may produce better HI. Also, a significant and critical point in the predefined function scenario to fuse features that should be seriously noted is the computation time. Since the extracted features might be more than hundreds (like features extracted from acoustic emission in the current work), this model is very time-consuming. Thus, a fusion paradigm based on an ANN rather than predefined functions is proposed in this thesis.

AI-based procedures to extract HIs

ANN and especially deep neural networks (DNN) are applicable in the field of PHM [74–76] and are powerful mathematical methods, having potential for approaching the HI construction problem. In 2003, the introduction of the minimum quantization error (MQE) index, derived from the self-organizing map (SOM) approach [77], marked a novel degradation indicator for bearings. The results showcased its superiority over the commonly used RMS as the typical HI for bearings. However, a quantitative comparison based on evaluation criteria was not provided since these criteria were not introduced until 2010 [50] for prognostic purposes. Nonetheless, based on their plots for three bearings, high criteria scores were not expected for all Mo, Pr, and Tr. It is worth noting that, as explained in the PHI subsection, RMS and RRMS are still acceptable HIs for bearings, as RMS was used as the comparison basis in that study to be outperformed. The prognostics of composite structures pose even greater challenges, lacking a foundational basis for comparison.

Later, the MQE index was further utilized and improved in subsequent works [78]. In 2017, this index, named "a self-organizing map-based HI (SOM-HI)," was used as the basis for comparison with a recurrent neural network-based health indicator (RNN-HI) for the RUL prediction of bearings in the most cited work regarding HI construction. The authors assumed a linearly increasing trend for the HI to feed targets into their model. Correlation and Mo were used as the evaluation criteria, resulting in an average score ($[Mo + Correlation]/2$) of 0.6871 for SOM-HI and 0.7471 for RNN-HI. The best feature input from the frequency domain had an average score of 0.5411. The results highlighted two key findings: (1) SOM-HI showed limited promise, with only ≈14% improvement compared to the input feature, evaluated based on Mo and Tr criteria (assuming that Correlation is equivalent to Tr, which is not). *Note that Tr measures the relevance between the degradation trends of HIs for a set of related units (e.g., bearings or composite structures) [36, 64, 79, 80], whereas the Correlation metric measures the relevance between a HI's degrada-*

tion trend and the unit's operating time, which implies an expectation of a linear relationship between the HI and operation time. (2) Neither SOM-HI nor RNN-HI demonstrated a linear trend. Thus, the assumption of a linear trend for the HI to generate model targets is questionable. Moreover, a nonlinear behavior is expected for HIs due to the stochastic nature of damage propagation and accumulation in any isotropic material—let alone in composite structures. Although Pr was not quantitatively reported and compared, the displayed figures indicated a significantly better Pr score for RNN-HI compared to SOM-HI. It is essential to note that the reported evaluation criteria were based on six "training" bearings, raising questions about the validity of the work to some extent. Another limitation was the selection of the best features based on criteria before subsequent steps.

Motivated by RNN-HI, additional works were conducted following the linear trend for the HI. In 2018, a convolutional neural network (CNN) was employed to formulate a HI (CNN-HI) with an outlier removal step [81]. Three criteria, named "Monotonicity, Trendability, and Scales," were utilized, where the first one was Mo, and the second one was the Correlation metric instead of Tr. The Scales metric was likely intended to represent Pr, but it might be misleading, as it considered the maximum HI value throughout the entire operational period instead of focusing on the HI value at EoL. The scores for Mo, Correlation, and Scales were reported as 0.406, 0.897, and 0.904, respectively. Considering the average score ($[Mo + Correlation]/2$) of 0.651, the results did not surpass RNN-HI. This could be attributed to the higher significance of temporal information than spatial information in the HI construction process. Building upon this, in another study conducted in 2020, a convolutional recurrent neural network (CRNN-HI) was created to construct a HI aligning with the linear trend [80]. The evaluation focused solely on Mo and Correlation metrics. The CRNN-HI demonstrated a slight enhancement compared to RNN-HI, attributed to the exploitation of spatial information in addition to temporal information. However, there was only a modest improvement, and the Mo scores remained relatively low.

The application of autoencoder has proven successful in diagnosing faults in rotating equipment [82, 83]. Chen et al. [82] introduced fused feature vectors as machine HIs for further classification using a deep belief network (DBN). It is important to note that their HI does not precisely align with the intended HI in this study; rather, it is just a fault indicator for damage classification and not for RUL prediction. Also, they employed max-min normalization, an unacceptable step in prognostics due to the unavailability of entire historical data up to the EoL. Similarly, a contractive autoencoder (CAE) was applied for gearbox diagnosis [83]. To enhance the capabilities of a stacked sparse autoencoder (SSAE), Qi et al. [84] integrated ensemble empirical mode decomposition (EEMD) and autoregression to preprocess the original vibration signal, selecting intrinsic mode function components (IMFs) from EEMD as inputs for SSAE in fault diagnosis.

In 2018, Yang et al. [85] utilized sparse autoencoders to construct HIs for bearings, considering only Mann-Kendall (MK) monotonicity, which provides a score not within the [0, 1] range, which makes it difficult for comparison. In 2019, Lin et al. [86] developed an ensemble stacked autoencoder (ESA) to construct HIs with linear targets. They extracted features from vibration frequency spectra using four stacked activation-diverse autoencoders, then fed these features into a DNN to reduce the dimensionality to one. Based on Mo, Correlation, and Robustness criteria, they achieved a weighted score ($0.35 \times Mo + 0.35 \times Correlation + 0.3 \times Robustness$) of 0.674 for one bearing, slightly lower

than EMDCNN-HI, a multi-channel deep convolutional neural network with an exponentially decaying learning rate [87], which scored 0.718. She et al. [88] applied a sparse autoencoder with a regularization network (SAEwR-HI) to compress raw features, constructing a HI for rolling bearings based on features with high Correlation. They obtained scores of 0.2826, 0.9940, and 0.9274 for Mo, Correlation, and Robustness, respectively, resulting in a weighted score ($0.35 \times Mo + 0.35 \times Correlation + 0.3 \times Robustness$) of 0.7250 for three bearings, surpassing autoencoder-HI (AE-HI), variational autoencoder-HI (VAE-HI), and PCA-HI. Xu et al. [89] employed stacked autoencoders to detect the initial degradation process through the frequency spectra of bearings. They improved Mo using an exponential function on the extracted HI from the stacked autoencoder curve. Unfortunately, they normalized inputs using max-min normalization, an unacceptable step for prognostic purposes.

The literature review highlights several research gaps and noteworthy observations:

- **Limited Applicability to Composite Structures:** The discussed works primarily focus on rotating mechanical equipment, particularly bearings, while a research gap has been identified in the field of structural applications, particularly in the construction of HIs for composite structures. This distinction is noteworthy, as the rotational dynamics inherent in rotating machinery act as an activator, accentuating defects in signals from faulty units compared to a healthy condition. In contrast, composite structures operate in a more quasi-static mode for micro- and macro-scale damage due to their loading frequency in comparison to rotating machinery applications. This quasi-static nature renders them more passive in monitoring, presenting a challenge for diagnosis and prognosis, particularly when using passive SHM methods like acoustic emission. Moreover, the inhomogeneous nature of composite structures introduces added complexity and stochasticity in damage initiation and growth.
- **Nonlinear Nature of Damage Propagation:** All of the studies cited above considered a linear function for HIs, while not only were nonlinear HIs obtained after training the models in all the cited studies, but also PHIs have so far been nonlinear. Thus, the assumption of a linear trend for HIs to generate targets for models is deemed questionable based on the physics of the problem, which involves nonlinear damage propagation and accumulation.
- **Limited Generalizability and Model Stability:** The lack of evaluation regarding generalizability is a noteworthy gap. The studies often investigate a small number of units (one to three), with limited reports on the stability of models considering cross-validation and enough replications. Particularly, DNNs, known for their randomness and strong sensitivity to initial weights and biases, face challenges for stability checks.
- **Unclear Criteria Evaluation:** The criteria used for evaluation are not consistently reported. It remains unclear whether criteria calculations were based on training units only, a combination of training and test units, or solely on test units. Additionally, there is a lack of criteria specifically evaluating HIs, considering only the test unit.

Addressing these gaps is crucial for advancing the understanding and applicability of HIs in diverse contexts, especially in composite structures which are different from rotating machinery equipment.

2.6 Interpretability vs. Explainability vs. Accuracy

The ML field commonly faces criticism for the lack of a clear definition of "interpretability." Furthermore, instead of a single ground-truth explanation, multiple quantifiable dimensions of interpretability are emerging [90]. The domain of interpretable ML (IML), often also referred to as explainable AI (XAI), has experienced significant growth in recent years [90]. Approaches to ML interpretability can be categorized based on various criteria: intrinsic or post-hoc. This classification distinguishes between methods achieving interpretability by constraining the complexity of the ML model (intrinsic) and those analyzing the model after training (post hoc) [91]. In this study, the focus lies on HI models and their application rather than general ML. Accordingly, keywords such as interpretability and explainability are redefined to better present different aspects of interpretability in HI construction models.

Although a HI is employed by a prognostic model to forecast the RUL, it can bring further value, such as interpretability and a closer connection to the component's health (damage) status. One aspect that cannot be addressed using prognostic criteria is the interpretability of a HI. In recent years, a number of data-driven models have been developed to extract HI [92, 93]. However, the HI functions produced by data-driven models are so complicated that they are almost beyond comprehension, i.e., they lose their physical meanings [94]. Even in the case of linear-based models, such as PCA-based techniques mentioned earlier, the PCs they produce cannot be straightforwardly interpreted as physical characteristics of the unit. This lack of interpretability can be a drawback in certain scenarios. In other words, each PC contains all of the inputs (which may not be needed), which are activated by a kernel (in the case of KPCA) and multiplied by different weights, causing complexity, and each PC has different coefficients compared to the others.

Sometimes it is even challenging to translate the raw data from a sensor into a physical phenomenon, let alone interpret the output of a complex data-driven model run on a network of various sensors. Thus, the first step towards understanding the physics of a unit could be made if the interpretability of the fusion model of the sensory inputs–i.e., understanding which sensors were used and how they formed the output (HI)–is possible.

As discussed in Chapter 1 (section 1.2), interpretability can be approached in two ways:

1. **Physics-based relationship with HI:** Finding a HI that exhibits a meaningful relationship with the physical and mechanical behaviors of the unit.

2. **Relationship between HI and SHM Data:** Understanding the relationship between the constructed HI and SHM data, assuming a prior comprehension of the relationship between various damage mechanisms in composite materials and SHM data, which can be categorized into two levels of understanding:

    - ***Interpretable HI:*** Grasping the precise mathematical function between the HI and SHM data.

- ***Explainable HI:*** Comprehending the correlation between the HI and SHM data.

The construction of HIs for composite structures is a new area compared to rotating machinery, demanding further research and development to reach maturity. Additionally, there is currently no established, promising PHI that serves as a benchmark meeting all prognostic criteria, posing a challenge for the first approach. Moreover, understanding the intricate connection between SHM data and diverse damage mechanisms in composite materials is a complex and evolving task [95–98], posing a challenge for the second approach. Nevertheless, as demonstrated in Appendix A.1, a methodology has been devised to design an interpretable HI (in the second approach) for commercial turbofan engines (CMAPSS dataset), which is more straightforward compared to datasets involving composite structures under cyclic fatigue loading. The developed model belongs to the intrinsic IML category. It should be noted that interpretability is not among the primary or secondary objectives of this thesis but other objectives, as outlined in Section 1.4.

In applications with massive amounts of data, DL algorithms can be utilized to automatically build HIs without requiring a lot of domain knowledge. On the other hand, due to the thousands or even millions of parameters required by an applicable ANN for generating HIs, the features produced by DL are complex to interpret and cannot be treated as physical characteristics of the unit under monitoring. In fact, data-driven approaches (such as ANN, reinforcement learning, etc.) offer little insight into the relation between the inputs of captured sensory data and the outputs (either HI or RUL) of the model (black-box) [99]. One of the main causes is that DL models typically have thousands of parameters [100], e.g., when generating a HI, which makes them less generalized and very complicated (the formula behind the DL model is not readable). This renders them inefficient in terms of interpretability.

Typical ANNs employ additive neurons, which multiply inputs by weights before summing the outputs. As a consequence, the option to multiply the inputs together is missing, particularly in situations where numerous inputs are involved, such as CM sensory data. Instead of only taking additive neurons into account, the multiplicative operator may produce a simpler, more inclusive, and more understandable equation. In fact, a multiplicative operator can supply multiple summation operators, resulting in a shorter length for the output equation. For instance, the HI function developed by [52], for the CMAPSS dataset, uses just the multiplication and division operators between the features—there is no use of the summing operator. In order to simulate multiplication and division operations using purely summing operators (assuming this is doable), it is most likely necessary to use more weighted summation operators, which would make the HI function more complicated to comprehend. Also, retaining continuous weights for each neuron still results in a complex and non-understandable equation, potentially containing many unnecessary terms that need removal. Given this, an approach is required to tackle these challenges in the context of interpretability.

In addition to the explanations provided above, achieving accurate results and high performance (e.g., meeting prognostic criteria in this book) for complex units requires not only continuous weights and different neurons, including additive and multiplicative ones alongside each other, but also deep layers to uncover the relationship between inputs and outputs. Consequently, even with applying all the aforementioned procedures

to a model, obtaining an interpretable yet accurate model seems impossible. For example, many researchers assert that all physics-based models of damage mechanics in composite materials are interpretable since they are derived from other basic, well-known physics-based equations, laws, models, etc., while the author argues with that notion.

For instance, considering Ladeveze's damage modeling for laminated composites [101], Ladeveze and Dantec introduced their theoretical modeling of the elementary ply, incorporating various physical concepts such as damage kinematics of the elementary ply, plasticity modeling, damage-plasticity coupling, and fiber-direction behavior modeling. While each physical concept and its equations may be understood, comprehending the entire model, including all these concepts and their interactions at the same time, is challenging. Moreover, this model is a 2D model, assuming a plane-stress state. Expanding it to a full 3D structure and incorporating other nonlinear factors for improved accuracy makes it even more complex. As shown in Figure 1.2, this physics-based damage model may not be deemed an end-to-end interpretable model, similar to a DNN model, where the equation of each neuron (its weights and activation function) is known, but the DNN model is considered a black box in the end-to-end view. Thus, two questions remain:

1. Is interpretability needed, and does it help? If yes, to what level of interpretability is needed?

2. What holds higher priority for a complicated engineering system or structure: interpretability, explainability, or accuracy?

2.7 Conclusions

The literature review has shed light on research gaps and challenges in the landscape of HIs, especially for composite structures. The upcoming chapters of the book will actively address these gaps, focusing on:

- **Limited Applicability to Composite Structures:** The thesis will explore and propose methodologies toward HI design specifically for composite structures.

- **Nonlinear HI:** Recognizing the nonlinear nature of damage propagation and health degradation, the thesis will advocate for and provide solutions to move beyond the conventional linear assumptions in constructing HIs.

- **Generalizability and Criteria Evaluation:** Through rigorous methodologies, the thesis will enhance the generalizability of models, employing thorough stability checks and transparent criteria evaluation procedures, especially focusing on test units.

- **Need for Historical-Independent HI:** Acknowledging the common historical-dependency drawback in HI construction and prognostic models, this work will propose a strategy to mitigate this drawback and create HIs for composite structures that are not reliant on historical SHM data.

- **Fusion of AE and GW:** The thesis will delve into fusing passive and active SHM methods, maximizing the benefits of both temporal and spatial information.

- **Interpretability:** AE- and GW-based frameworks were investigated in terms of interpretability, where the presence of incremental steps in the generated HIs may correspond to distinct damage states within composite structures. Additionally, a methodology for designing interpretable HIs for commercial turbofan engines (the CMAPSS dataset) will be developed in this research, making HIs more understandable as well as transparent and contributing to the interpretability aspect of HIs.

In summary, the upcoming chapters strive to address these gaps while also introducing innovative methodologies and insights, contributing to the progression of the field of HIs for composite structures.

References

[1] P. W. Beaumont, "The structural integrity of composite materials and long-life implementation of composite structures," *Applied Composite Materials*, vol. 27, no. 5, pp. 449–478, 2020.

[2] S. Hassani, M. Mousavi, and A. H. Gandomi, "Structural health monitoring in composite structures: A comprehensive review," *Sensors*, vol. 22, no. 1, p. 153, 2021.

[3] M. Senthilkumar, T. Sreekanth, and S. Manikanta Reddy, "Nondestructive health monitoring techniques for composite materials: A review," *Polymers and Polymer Composites*, vol. 29, no. 5, pp. 528–540, 2021.

[4] W. Nsengiyumva, S. Zhong, J. Lin, Q. Zhang, J. Zhong, and Y. Huang, "Advances, limitations and prospects of nondestructive testing and evaluation of thick composites and sandwich structures: A state-of-the-art review," *Composite Structures*, vol. 256, p. 112951, 2021.

[5] P. Zhang, Y. Feng, T. Q. Bui, X. Hu, and W. Yao, "Modelling distinct failure mechanisms in composite materials by a combined phase field method," *Composite Structures*, vol. 232, p. 111551, 2020.

[6] K. Venkatesan, T. Stoumbos, D. Inoyama, and A. Chattopadhyay, "Computational analysis of failure mechanisms in composite sandwich space structures subject to cyclic thermal loading," *Composite Structures*, vol. 256, p. 113086, 2021.

[7] A. Güemes, A. Fernandez-Lopez, A. R. Pozo, and J. Sierra-Pérez, "Structural health monitoring for advanced composite structures: A review," *Journal of Composites Science*, vol. 4, no. 1, p. 13, 2020.

[8] X. Jia and X. Chen, "Ai-based optical-thermal video data fusion for near real-time blade segmentation in normal wind turbine operation," *Engineering Applications of Artificial Intelligence*, vol. 127, p. 107325, 2024.

[9] K. Askaripour and A. Zak, "A survey of scrutinizing delaminated composites via various categories of sensing apparatus," *Journal of Composites Science*, vol. 3, no. 4, p. 95, 2019.

[10] S. Hassani and U. Dackermann, "A systematic review of advanced sensor technologies for non-destructive testing and structural health monitoring," *Sensors*, vol. 23, no. 4, p. 2204, 2023.

[11] N. Ida and N. Meyendorf, *Handbook of advanced nondestructive evaluation*, vol. 10. Springer International Publishing Cham, Switzerland, 2019.

[12] N. Beganovic and D. Söffker, "Structural health management utilization for lifetime prognosis and advanced control strategy deployment of wind turbines: An overview and outlook concerning actual methods, tools, and obtained results," *Renewable and Sustainable Energy Reviews*, vol. 64, pp. 68–83, 2016.

[13] Y. Du, S. Zhou, X. Jing, Y. Peng, H. Wu, and N. Kwok, "Damage detection techniques for wind turbine blades: A review," *Mechanical Systems and Signal Processing*, vol. 141, p. 106445, 2020.

[14] R. De Oliveira and A. Marques, "Health monitoring of frp using acoustic emission and artificial neural networks," *Computers & structures*, vol. 86, no. 3-5, pp. 367–373, 2008.

[15] X. Li, M. Saeedifar, R. Benedictus, and D. Zarouchas, "Damage accumulation analysis of cfrp cross-ply laminates under different tensile loading rates," *Composites Part C: Open Access*, vol. 1, p. 100005, 2020.

[16] M. Saeedifar, M. Fotouhi, R. Mohammadi, M. Ahmadi, and M. Hajikhani, "Classification of damage mechanisms during delamination growth in sandwich composites by acoustic emission," *Modares Mechanical Engineering*, vol. 14, no. 6, pp. 144–152, 2014.

[17] M. Saeedifar, M. Ahmadi Najafabadi, K. Mohammadi, M. Fotouhi, H. Hosseini Toudeshky, and R. Mohammadi, "Acoustic emission-based methodology to evaluate delamination crack growth under quasi-static and fatigue loading conditions," *Journal of nondestructive evaluation*, vol. 37, pp. 1–13, 2018.

[18] R. Lima, M. Drobiazko, A. Bernasconi, and M. Carboni, "On crack tip localisation in quasi-statically loaded, adhesively bonded double cantilever beam specimens by acoustic emission," *Theoretical and Applied Fracture Mechanics*, vol. 118, p. 103286, 2022.

[19] M. Saeedifar and D. Zarouchas, "Damage characterization of laminated composites using acoustic emission: A review," *Composites Part B: Engineering*, vol. 195, p. 108039, 2020.

[20] B. Zhang, D. Yang, X. Hong, and G. Jin, "Deep emulational semi-supervised knowledge probability imaging method for plate structural health monitoring using guided waves," *Engineering with Computers*, vol. 38, no. 5, pp. 4151–4166, 2022.

[21] L. Tang, Y. Li, Q. Bao, W. Hu, Q. Wang, Z. Su, and D. Yue, "Quantitative identification of damage in composite structures using sparse sensor arrays and multi-domain-feature fusion of guided waves," *Measurement*, vol. 208, p. 112482, 2023.

[22] Z. Su, L. Ye, and Y. Lu, "Guided lamb waves for identification of damage in composite structures: A review," *Journal of sound and vibration*, vol. 295, no. 3-5, pp. 753–780, 2006.

[23] A. De Luca, F. Caputo, Z. S. Khodaei, and M. Aliabadi, "Damage characterization of composite plates under low velocity impact using ultrasonic guided waves," *Composites Part B: Engineering*, vol. 138, pp. 168–180, 2018.

[24] M. Ahmadivala, *Towards optimal maintenance planning of existing structures based on time-dependent reliability analysis*. PhD thesis, Université Clermont Auvergne [2017-2020], 2020.

[25] A. J. Valkonen, *Exploring Structural Health Monitoring Value of Information Based on Remaining Useful Life Extension Potential*. PhD thesis, Princeton University, 2023.

[26] P. Kamranfar, *Machine Learning Enabled Health Monitoring And Diagnosis of Engineering Systems*. PhD thesis, George Mason University, 2023.

[27] J.-B. Ihn and F.-K. Chang, "Pitch-catch active sensing methods in structural health monitoring for aircraft structures," *Structural Health Monitoring*, vol. 7, no. 1, pp. 5–19, 2008.

[28] V. Memmolo, E. Monaco, N. Boffa, L. Maio, and F. Ricci, "Guided wave propagation and scattering for structural health monitoring of stiffened composites," *Composite Structures*, vol. 184, pp. 568–580, 2018.

[29] R. Gorgin, Y. Luo, and Z. Wu, "Environmental and operational conditions effects on lamb wave based structural health monitoring systems: A review," *Ultrasonics*, vol. 105, p. 106114, 2020.

[30] G. B. Santoni, L. Yu, B. Xu, and V. Giurgiutiu, "Lamb wave-mode tuning of piezoelectric wafer active sensors for structural health monitoring," 2007.

[31] H. Liu, S. Liu, Z. Liu, N. Mrad, and A. S. Milani, "Data-driven approaches for characterization of delamination damage in composite materials," *IEEE Transactions on Industrial Electronics*, vol. 68, no. 3, pp. 2532–2542, 2020.

[32] M. Rautela, J. Senthilnath, E. Monaco, and S. Gopalakrishnan, "Delamination prediction in composite panels using unsupervised-feature learning methods with wavelet-enhanced guided wave representations," *Composite Structures*, vol. 291, p. 115579, 2022.

[33] R. Miorelli, C. Fisher, A. Kulakovskyi, B. Chapuis, O. Mesnil, and O. D'Almeida, "Defect sizing in guided wave imaging structural health monitoring using convolutional neural networks," *NDT & E International*, vol. 122, p. 102480, 2021.

[34] A. Tabatabaeian, B. Jerkovic, P. Harrison, E. Marchiori, and M. Fotouhi, "Barely visible impact damage detection in composite structures using deep learning networks with varying complexities," *Composites Part B: Engineering*, vol. 264, p. 110907, 2023.

[35] M. Moradi, J. Chiachío, and D. Zarouchas, "Developing health indicators for composite structures based on a two-stage semi-supervised machine learning model using acoustic emission data," in *Proceedings of the 10th ECCOMAS Thematic Conference on Smart Structures and Materials*, vol. 10, 2023.

[36] M. Moradi, A. Broer, J. Chiachío, R. Benedictus, T. H. Loutas, and D. Zarouchas, "Intelligent health indicator construction for prognostics of composite structures utilizing a semi-supervised deep neural network and shm data," *Engineering Applications of Artificial Intelligence*, vol. 117, p. 105502, 2023.

[37] J. Contreras Lopez, J. Chiachío, A. Saleh, M. Chiachío, and A. Kolios, "A cross-sectoral review of the current and potential maintenance strategies for composite structures," *SN Applied Sciences*, vol. 4, no. 6, p. 180, 2022.

[38] J. J. M. Jimenez, S. Schwartz, R. Vingerhoeds, B. Grabot, and M. Salaün, "Towards multi-model approaches to predictive maintenance: A systematic literature survey on diagnostics and prognostics," *Journal of manufacturing systems*, vol. 56, pp. 539–557, 2020.

[39] L. Guo, N. Li, F. Jia, Y. Lei, and J. Lin, "A recurrent neural network based health indicator for remaining useful life prediction of bearings," *Neurocomputing*, vol. 240, pp. 98–109, 2017.

[40] G. Galanopoulos, D. Milanoski, A. Broer, D. Zarouchas, and T. Loutas, "Health monitoring of aerospace structures utilizing novel health indicators extracted from complex strain and acoustic emission data," *Sensors*, vol. 21, no. 17, p. 5701, 2021.

[41] A. Khan, M. M. Azad, M. Sohail, and H. S. Kim, "A review of physics-based models in prognostics and health management of laminated composite structures," *International Journal of Precision Engineering and Manufacturing-Green Technology*, pp. 1–21, 2023.

[42] A. Al-Dulaimi, S. Zabihi, A. Asif, and A. Mohammadi, "A multimodal and hybrid deep neural network model for remaining useful life estimation," *Computers in industry*, vol. 108, pp. 186–196, 2019.

[43] Z. Chen, H. Zhu, J. Wu, and L. Fan, "Health indicator construction for degradation assessment by embedded lstm–cnn autoencoder and growing self-organized map," *Knowledge-Based Systems*, vol. 252, p. 109399, 2022.

[44] M. Moradi, P. Komninos, R. Benedictus, and D. Zarouchas, "Interpretable neural network with limited weights for constructing simple and explainable hi using shm data," in *Annual Conference of the PHM Society*, vol. 14, 2022.

[45] C. Song, K. Liu, and X. Zhang, "Integration of data-level fusion model and kernel methods for degradation modeling and prognostic analysis," *IEEE Transactions on Reliability*, vol. 67, no. 2, pp. 640–650, 2017.

[46] Y. Lei, N. Li, L. Guo, N. Li, T. Yan, and J. Lin, "Machinery health prognostics: A systematic review from data acquisition to rul prediction," *Mechanical systems and signal processing*, vol. 104, pp. 799–834, 2018.

[47] C. Hu, B. D. Youn, P. Wang, and J. T. Yoon, "Ensemble of data-driven prognostic algorithms for robust prediction of remaining useful life," *Reliability Engineering & System Safety*, vol. 103, pp. 120–135, 2012.

[48] A. Saxena, J. Celaya, E. Balaban, K. Goebel, B. Saha, S. Saha, and M. Schwabacher, "Metrics for evaluating performance of prognostic techniques," in *2008 international conference on prognostics and health management*, pp. 1–17, IEEE, 2008.

[49] Y. Lei, *Intelligent fault diagnosis and remaining useful life prediction of rotating machinery*. Butterworth-Heinemann, 2016.

[50] J. B. Coble, "Merging data sources to predict remaining useful life–an automated method to identify prognostic parameters," 2010.

[51] L. Saidi, J. B. Ali, E. Bechhoefer, and M. Benbouzid, "Wind turbine high-speed shaft bearings health prognosis through a spectral kurtosis-derived indices and svr," *Applied Acoustics*, vol. 120, pp. 1–8, 2017.

[52] K. T. Nguyen and K. Medjaher, "An automated health indicator construction methodology for prognostics based on multi-criteria optimization," *ISA transactions*, vol. 113, pp. 81–96, 2021.

[53] M. L. Baptista, K. Goebel, and E. M. Henriques, "Relation between prognostics predictor evaluation metrics and local interpretability shap values," *Artificial Intelligence*, vol. 306, p. 103667, 2022.

[54] F. Cadini, C. Sbarufatti, F. Cancelliere, and M. Giglio, "State-of-life prognosis and diagnosis of lithium-ion batteries by data-driven particle filters," *Applied energy*, vol. 235, pp. 661–672, 2019.

[55] H. Pei, X.-S. Si, C. Hu, T. Li, C. He, and Z. Pang, "Bayesian deep-learning-based prognostic model for equipment without label data related to lifetime," *IEEE Transactions on Systems, Man, and Cybernetics: Systems*, vol. 53, no. 1, pp. 504–517, 2022.

[56] G. Huang, S. Hua, Q. Zhou, H. Li, and Y. Zhang, "Just another attention network for remaining useful life prediction of rolling element bearings," *Ieee Access*, vol. 8, pp. 204144–204152, 2020.

[57] Y. Zhang, J. Sun, J. Zhang, H. Shen, Y. She, and Y. Chang, "Health state assessment of bearing with feature enhancement and prediction error compensation strategy," *Mechanical Systems and Signal Processing*, vol. 182, p. 109573, 2023.

[58] N. Eleftheroglou and T. Loutas, "Fatigue damage diagnostics and prognostics of composites utilizing structural health monitoring data and stochastic processes," *Structural Health Monitoring*, vol. 15, no. 4, pp. 473–488, 2016.

[59] T. Li, C. Sbarufatti, F. Cadini, J. Chen, and S. Yuan, "Particle filter-based hybrid damage prognosis considering measurement bias," *Structural Control and Health Monitoring*, vol. 29, no. 4, p. e2914, 2022.

[60] G. Seon, A. Makeev, J. Cline, and B. Shonkwiler, "Assessing 3d shear stress–strain properties of composites using digital image correlation and finite element analysis based optimization," *Composites science and technology*, vol. 117, pp. 371–378, 2015.

[61] M. Saeedifar, M. N. Saleh, P. Nijhuis, S. T. de Freitas, and D. Zarouchas, "Damage assessment of a titanium skin adhesively bonded to carbon fiber–reinforced plastic omega stringers using acoustic emission," *Structural Health Monitoring*, vol. 21, no. 2, pp. 407–423, 2022.

[62] C. Qian, T. Westphal, and R. Nijssen, "Micro-mechanical fatigue modelling of unidirectional glass fibre reinforced polymer composites," *Computational materials science*, vol. 69, pp. 62–72, 2013.

[63] B. Ameri, M. Moradi, B. Mohammadi, and D. Salimi-Majd, "Investigation of nonlinear post-buckling delamination in curved laminated composite panels via cohesive zone model," *Thin-Walled Structures*, vol. 154, p. 106797, 2020.

[64] N. Eleftheroglou, "Adaptive prognostics for remaining useful life of composite structures," 2020.

[65] Y. Qin, S. Xiang, Y. Chai, and H. Chen, "Macroscopic–microscopic attention in lstm networks based on fusion features for gear remaining life prediction," *IEEE Transactions on Industrial Electronics*, vol. 67, no. 12, pp. 10865–10875, 2019.

[66] T. Song, C. Liu, R. Wu, Y. Jin, and D. Jiang, "A hierarchical scheme for remaining useful life prediction with long short-term memory networks," *Neurocomputing*, vol. 487, pp. 22–33, 2022.

[67] G. Chen, J. Chen, Y. Zi, J. Pan, and W. Han, "An unsupervised feature extraction method for nonlinear deterioration process of complex equipment under multi dimensional no-label signals," *Sensors and Actuators A: Physical*, vol. 269, pp. 464–473, 2018.

[68] A. Thieullen, M. Ouladsine, and J. Pinaton, "A survey of health indicators and data-driven prognosis in semiconductor manufacturing process," *IFAC Proceedings Volumes*, vol. 45, no. 20, pp. 19–24, 2012.

[69] L. Cao, K. S. Chua, W. Chong, H. Lee, and Q. Gu, "A comparison of pca, kpca and ica for dimensionality reduction in support vector machine," *Neurocomputing*, vol. 55, no. 1-2, pp. 321–336, 2003.

[70] P. Wen, S. Zhao, S. Chen, and Y. Li, "A generalized remaining useful life prediction method for complex systems based on composite health indicator," *Reliability Engineering & System Safety*, vol. 205, p. 107241, 2021.

[71] D. Chen, Y. Qin, Y. Wang, and J. Zhou, "Health indicator construction by quadratic function-based deep convolutional auto-encoder and its application into bearing rul prediction," *ISA transactions*, vol. 114, pp. 44–56, 2021.

[72] B. Zhang, L. Zhang, and J. Xu, "Degradation feature selection for remaining useful life prediction of rolling element bearings," *Quality and Reliability Engineering International*, vol. 32, no. 2, pp. 547–554, 2016.

[73] N. Eleftheroglou, D. Zarouchas, T. Loutas, R. Alderliesten, and R. Benedictus, "Structural health monitoring data fusion for in-situ life prognosis of composite structures," *Reliability Engineering & System Safety*, vol. 178, pp. 40–54, 2018.

[74] S. Khan and T. Yairi, "A review on the application of deep learning in system health management," *Mechanical Systems and Signal Processing*, vol. 107, pp. 241–265, 2018.

[75] R. Zhao, R. Yan, Z. Chen, K. Mao, P. Wang, and R. X. Gao, "Deep learning and its applications to machine health monitoring," *Mechanical Systems and Signal Processing*, vol. 115, pp. 213–237, 2019.

[76] O. Fink, Q. Wang, M. Svensen, P. Dersin, W.-J. Lee, and M. Ducoffe, "Potential, challenges and future directions for deep learning in prognostics and health management applications," *Engineering Applications of Artificial Intelligence*, vol. 92, p. 103678, 2020.

[77] H. Qiu, J. Lee, J. Lin, and G. Yu, "Robust performance degradation assessment methods for enhanced rolling element bearing prognostics," *Advanced Engineering Informatics*, vol. 17, no. 3-4, pp. 127–140, 2003.

[78] R. Huang, L. Xi, X. Li, C. R. Liu, H. Qiu, and J. Lee, "Residual life predictions for ball bearings based on self-organizing map and back propagation neural network methods," *Mechanical systems and signal processing*, vol. 21, no. 1, pp. 193–207, 2007.

[79] P. Nectoux, R. Gouriveau, K. Medjaher, E. Ramasso, B. Chebel-Morello, N. Zerhouni, and C. Varnier, "Pronostia: An experimental platform for bearings accelerated degradation tests.," in *IEEE International Conference on Prognostics and Health Management, PHM'12.*, pp. 1–8, IEEE Catalog Number: CPF12PHM-CDR, 2012.

[80] L. Chen, G. Xu, S. Zhang, W. Yan, and Q. Wu, "Health indicator construction of machinery based on end-to-end trainable convolution recurrent neural networks," *Journal of Manufacturing Systems*, vol. 54, pp. 1–11, 2020.

[81] L. Guo, Y. Lei, N. Li, T. Yan, and N. Li, "Machinery health indicator construction based on convolutional neural networks considering trend burr," *Neurocomputing*, vol. 292, pp. 142–150, 2018.

[82] Z. Chen and W. Li, "Multisensor feature fusion for bearing fault diagnosis using sparse autoencoder and deep belief network," *IEEE Transactions on Instrumentation and Measurement*, vol. 66, no. 7, pp. 1693–1702, 2017.

[83] C. Shen, Y. Qi, J. Wang, G. Cai, and Z. Zhu, "An automatic and robust features learning method for rotating machinery fault diagnosis based on contractive autoencoder," *Engineering Applications of Artificial Intelligence*, vol. 76, pp. 170–184, 2018.

[84] Y. Qi, C. Shen, D. Wang, J. Shi, X. Jiang, and Z. Zhu, "Stacked sparse autoencoder-based deep network for fault diagnosis of rotating machinery," *Ieee Access*, vol. 5, pp. 15066–15079, 2017.

[85] Z. Yang, P. Baraldi, and E. Zio, "Automatic extraction of a health indicator from vibrational data by sparse autoencoders," in *2018 3rd International Conference on System Reliability and Safety (ICSRS)*, pp. 328–332, IEEE, 2018.

[86] P. Lin and J. Tao, "A novel bearing health indicator construction method based on ensemble stacked autoencoder," in *2019 IEEE international conference on prognostics and health management (ICPHM)*, pp. 1–9, IEEE, 2019.

[87] D. She and M. Jia, "Wear indicator construction of rolling bearings based on multi-channel deep convolutional neural network with exponentially decaying learning rate," *Measurement*, vol. 135, pp. 368–375, 2019.

[88] D. She, M. Jia, and M. G. Pecht, "Sparse auto-encoder with regularization method for health indicator construction and remaining useful life prediction of rolling bearing," *Measurement Science and Technology*, vol. 31, no. 10, p. 105005, 2020.

[89] F. Xu, Z. Huang, F. Yang, D. Wang, and K. L. Tsui, "Constructing a health indicator for roller bearings by using a stacked auto-encoder with an exponential function to eliminate concussion," *Applied Soft Computing*, vol. 89, p. 106119, 2020.

[90] C. Molnar, G. Casalicchio, and B. Bischl, "Interpretable machine learning–a brief history, state-of-the-art and challenges," in *Joint European conference on machine learning and knowledge discovery in databases*, pp. 417–431, Springer, 2020.

[91] C. Molnar, *Interpretable machine learning*. Lulu. com, 2020.

[92] D. Chen, Y. Qin, Q. Qian, Y. Wang, and F. Liu, "Transfer life prediction of gears by cross-domain health indicator construction and multi-hierarchical long-term memory augmented network," *Reliability Engineering & System Safety*, vol. 230, p. 108916, 2023.

[93] I. de Pater and M. Mitici, "Developing health indicators and rul prognostics for systems with few failure instances and varying operating conditions using a lstm autoencoder," *Engineering Applications of Artificial Intelligence*, vol. 117, p. 105582, 2023.

[94] Q. Ni, J. Ji, and K. Feng, "Data-driven prognostic scheme for bearings based on a novel health indicator and gated recurrent unit network," *IEEE Transactions on Industrial Informatics*, vol. 19, no. 2, pp. 1301–1311, 2022.

[95] M. Saeedifar and M. N. Saleh, "The effect of alternating the sequence of variable-energy repeated impact on the residual strength and damage evolution of composite laminates," *Polymer Composites*, vol. 44, no. 8, pp. 5029–5043, 2023.

[96] M. Saeedifar, "Impact damage assessment in laminated composites using acoustic emission and finite element methods," *Polymer Composites*, 2023.

[97] M. Farajpour and M. A. Najafabadi, "Investigation of crack propagation in filament wound composite samples of mode-i by using acoustic emission technique," *Journal of Composite Materials*, vol. 57, no. 25, pp. 3979–3989, 2023.

[98] S. Fotouhi, M. Assaad, M. Nasor, A. Imran, A. Ashames, and M. Fotouhi, "Multi-variable signal processing for characterization of failure modes in thin-ply hybrid laminates using acoustic emission sensors," *Sensors*, vol. 23, no. 11, p. 5244, 2023.

[99] M. Mitici, I. de Pater, A. Barros, and Z. Zeng, "Dynamic predictive maintenance for multiple components using data-driven probabilistic rul prognostics: The case of turbofan engines," *Reliability Engineering & System Safety*, vol. 234, p. 109199, 2023.

[100] J. Yan, Z. He, and S. He, "A deep learning framework for sensor-equipped machine health indicator construction and remaining useful life prediction," *Computers & Industrial Engineering*, vol. 172, p. 108559, 2022.

[101] P. Ladeveze and E. LeDantec, "Damage modelling of the elementary ply for laminated composites," *Composites science and technology*, vol. 43, no. 3, pp. 257–267, 1992.

3

Methodology

The chapter enhances HIs' evaluation criteria for test units in PHM, introduces a nonlinear HI design more aligned with damage physics, and concludes with an interpretable model for HI insights.

3.1 Introduction

As discussed in Chapter 2, developing a nonlinear HI is imperative to align with the physics of damage propagation and accumulation. Furthermore, the HIs' criteria (Mo, Pr, and Tr) are envisioned to play a supervising role, guiding the construction process rather than merely serving as a measurement tool for HI's quality. The upcoming chapter introduces and delves into these criteria, emphasizing their importance in evaluating HIs.

To enhance generalizability and criteria evaluation, we refine the approach by updating the criteria to specifically assess the performance on test units, which are crucial for real-world applications. The narrative then transitions to proposing a scenario for embedding the criteria in the fusion model. A novel semi-supervised (SS) paradigm is proposed to consider the HIs' criteria as supervisors. This paradigm, as detailed in subsequent chapters, exhibits versatility across various applications and model architectures, leveraging nonlinear kernels during the learning process.

Moreover, the HIs' criteria are explicitly implemented to optimize hyperparameters of the aforementioned SS models and in the ensemble learning step. The specifics of these implementations and developments will be elucidated in the next chapters, where each will be used.

Finally, the chapter ventures into the development of an interpretable model, aiming to contribute valuable insights into the interpretability aspect of HIs.

3.2 HIs' Criteria

Due to the high importance of HIs' evaluation metrics amongst different stages of the methodology, they are introduced at the outset. This section begins by introducing the prognostic metrics, which will be modified in the present work for consideration of only test units rather than both training and test units.

3.2.1 HIs' criteria given all units

The evaluation standard of HIs is based upon three established criteria, namely Mo, Pr, and Tr [1, 2]:

- The general increasing or decreasing pattern of a feature or generally a signal over time is expressed by Mo.
- The distribution of a parameter's failure (final) value is measured by Pr.
- The term Tr refers to whether a parameter's decay histories (degradation) have the same underlying pattern for different samples or systems under monitoring.

These metrics are defined as follows:

$$Mo = \frac{1}{M}\sum_{j=1}^{M}\left|\frac{1}{N_j-1}\sum_{i=1}^{N_j}\frac{\sum_{p=1,p>i}^{N_j}\left(t_p-t_i\right).sgn\left(x(t_p)-x(t_i)\right)}{\sum_{p=1,p>i}^{N_j}\left(t_p-t_i\right)}\right|.100\% \tag{3.1}$$

$$Pr = exp\left(-\frac{\sqrt{\frac{1}{M}\sum_{j=1}^{M}\left|x_j\left(N_j\right)-\left[\frac{1}{M}\sum_{i=1}^{M}x_i(N_i)\right]\right|^2}}{\frac{1}{M}\sum_{j=1}^{M}\left|x_j(1)-x_j\left(N_j\right)\right|}\right) \tag{3.2}$$

$$Tr = min_{j,k}\left|\frac{cov\left(x_j, x_k\right)}{\sigma_{x_j}\sigma_{x_k}}\right|, \quad j,\ k = 1,\ 2,\ \ldots, M \tag{3.3}$$

where $x(t_p)$ and $x(t_i)$ denote the measurements (HIs in the context) at the times of t_p and t_i, respectively. The $sgn(\bullet)$ function represents the signum function. $cov(x_j, x_k)$ signifies the covariance, where x_j and x_k are vectors of measurements for the j^{th} and k^{th} unit (out of M units – engineering systems or structures) with N_j and N_k measurements, respectively. The standard deviations of x_j and x_k are denoted by σ_{x_j} and σ_{x_k}, respectively. The evaluation metric selected for Mo in Eq. 3.1 is the Modified Mann-Kendall (MMK) metric. In comparison to the Sign and Mann-Kendall versions, MMK is more resilient to noise and considers the correlation between data points with time gaps exceeding one unit [1, 3]. All three HIs metrics (Mo, Pr, and Tr) are rated on a scale ranging from 0 to 1, where a score of 1 signifies optimal HI performance.

With consideration to these criteria, the Fitness metric is formulated as follows:

$$Fitness = a \cdot Mo_{HI} + b \cdot Pr_{HI} + c \cdot Tr_{HI} \tag{3.4}$$

Assuming the control constants a, b, and c each equal 1, the Fitness metric spans from 0 (indicating minimal quality) to 3 (indicating maximal quality) for the evaluated HIs.

3.2.2 HIs' criteria given test units

It is important to emphasize that the HIs' evaluation metrics mentioned above are devised to encompass all units under monitoring, specifically from their healthy state to their final failure status within the context of PHM. Without access to complete trajectories of HIs across all units, the assessment of HIs' quality lacks implication. Consequently, whether during the training or testing phase of ML-based models, the inclusion of all units becomes crucial to accurately measuring Fitness. However, a potential challenge arises due to the possibility of highly matched HIs during the training phase, which might result in a misleadingly high Fitness score when confronted with an unmatched (outlier) HI from a specific unit during the testing phase. Therefore, it is essential to evaluate HI metrics solely based on the test unit.

The choice of using the singular noun "test unit" instead of the plural "test units" is a reflection of the specific context within this work. In this study, we are focused on a single unit (composite specimen) designated for testing the model. This decision is dictated by the limited number of units available, with only 12 units (3 to 5 units for GW datasets used in Chapter 6) allocated for training and testing in the sequence-to-sequence (seq2seq) problem. However, it is noteworthy that similar functions, as will be briefly explained for each, could be readily extended to accommodate multiple test units.

In this study, the HIs' metrics, particularly Pr, are refined to focus more on the test units rather than incorporating the entire set (train/validation/test) of units. This adjustment is aimed at preventing the false positive influence of training units on final scores, which might overshadow the low scores of the test unit. By primarily considering the test unit, these updated metrics aim to determine if it deviates significantly from the training units and to what extent. It is noteworthy that if the metrics yield higher values for only the test unit while remaining lower for the training units, the methodology could be rendered

ineffective. This is because the models are trained on the training units, and if their HIs are not monotonic or if their Pr is low, high Mo and Pr for test units are not beneficial in the models' discrimination because the models have already observed other patterns. Conversely, if the metrics are consistently high for the training units, then achieving high scores for the test unit becomes meaningful.

The definition of Mo remains consistent, and its computation for a single unit follows a straightforward process involving the internal summation of Eq. 3.1:

$$Mo^{\tau} = \left| \frac{1}{N_{\tau}-1} \sum_{i=1}^{N_{\tau}} \frac{\sum_{p=1,p>i}^{N_{\tau}} \left(t_p - t_i\right).sgn\left(x\left(t_p\right) - x\left(t_i\right)\right)}{\sum_{p=1,p>i}^{N_{\tau}} \left(t_p - t_i\right)} \right| .100\% \tag{3.5}$$

where the symbol τ represents the test unit. In cases where multiple units are being considered for testing, Eq. 3.5 will be similar to Eq. 3.1, with the difference being the inclusion of an averaging procedure across only the test units.

However, Pr requires redefinition. In this adaptation, rather than considering the standard deviation of HIs at EoL across all units in the numerator, the deviation of the HI at EoL for the test unit from the deviation basis (i.e., its corresponding value averaged over the training units) is computed:

$$Pr^{\tau} = exp\left(-\frac{\left| x_{\tau}\left(N_{\tau}\right) - \overbrace{\left[\frac{1}{M^{\tau'}} \sum_{i=1}^{M^{\tau'}} x_i(N_i)\right]}^{\textbf{deviation basis}} \right|}{\underbrace{\frac{1}{M} \sum_{j=1}^{M} \left| x_j(1) - x_j\left(N_j\right) \right|}_{\textbf{scaling factor}}} \right) \tag{3.6}$$

where τ' represents all units set except for the test ones and $M^{\tau'}$ signifies the count of those units, including the training ones (or even the validation ones). It should be noted that $x_j(1)$ and $x_j\left(N_j\right)$ denote the HI values of the j^{th} unit at the initiation and EoL, respectively. The denominator serves as a scaling factor, which in this case corresponds to the mean value of the difference between HIs at the beginning and EoL across all units ($\tau \cup \tau'$) or the training units (τ'). After evaluating both options within this study, the mean value over all units is considered, as symbolized in Eq. 3.6. When dealing with multiple units (M^{τ}) under consideration for testing, the deviation basis of EoL could be established based on the training set (τ'), the test set (τ), or a combination of both ($\tau \cup \tau'$), among which the latest with M units is advisable:

$$Pr^{\tau} = exp\left(-\frac{\sqrt{\frac{1}{M^{\tau}}\sum_{j=1}^{M^{\tau}}\left|x_j\left(N_j\right) - \overbrace{\left[\frac{1}{M}\sum_{i=1}^{M}x_i(N_i)\right]}^{\textbf{deviation basis}}\right|^2}}{\underbrace{\frac{1}{M}\sum_{j=1}^{M}\left|x_j(1) - x_j\left(N_j\right)\right|}_{\textbf{scaling factor}}}\right) \tag{3.7}$$

Regarding Tr, it is crucial to note that the minimum correlation of HIs should be computed between two distinct units, which is not feasible when considering only a single test unit while excluding the training units. Additionally, if the correlation between the HI of the test unit and the HIs of the training units is calculated in a pairwise manner, followed by selecting the minimum as Tr, this value might surpass the correlation computed when considering all units' HIs pairwise. Consequently, to ensure a more stringent evaluation, the same formula as Eq. 3.3 is again applied to the test units. This approach is more rigorous and maintains consistency.

With considering the updated metrics, the Fitness metric for the test unit is as follows:

$$Fitness^{\tau} = a \cdot Mo_{HI}^{\tau} + b \cdot Pr_{HI}^{\tau} + c \cdot Tr_{HI} \tag{3.8}$$

where Tr_{HI} is the same as before.

3.3 Deviation from Simulated Ideal HIs

In addition to the evaluation criteria for HIs, a supplementary metric can be employed to gauge the deviation from the ideal HIs simulated under the semi-supervised learning (SSL) paradigm, as elucidated in the upcoming subsection. The root mean square error (RMSE) stands out as a suitable measure in this context:

$$RMSE = \sqrt{\frac{1}{N_j}\sum_{i=1}^{N_j}(T_i - HI_i)^2}, \quad j \in M\,, \quad p \in 1,\ 2 \tag{3.9}$$

In this formula, N_j is the length of the sequence (HI), T_i represents the target value (simulated HI under SSL paradigm), and HI_i is the network's output for time step i. The RMSE provides a single score for each unit's constructed HI, with a primary focus on reporting the test unit's HI score, excluding training or validation data.

3.4 Semi-Supervised Criteria-based Fusion Paradigm

In order to embed the Fitness function representing the criteria, different overall scenarios can be proposed. The simplest way is to rank the features according to their Fitness values. Although the feature with the highest rank can be considered as the HI, a threshold for the Fitness value can be set to accept more than one feature, and then, they can be considered as a set of HIs which are imported into a prognostic model. Otherwise, some

simple methods such as a weighted averaging can be applied to the features filtered by the threshold, and finally, only one HI can be exported. Nevertheless, such approach might result in the overlooking of useful features since they do not fit the criteria. However, their combination may fulfil the intended specifications. Thus, as earlier mentioned, a fusion model based on DNN and SSL, rather than the predefined functions, is proposed in the current study.

SSL enables enormous volumes of unlabeled data to be exploited in conjunction with normally smaller labeled data sets [4]. Unlabeled data can contribute to the formulation of a superior classifier or regressor, provided enough unlabeled data is available and certain assumptions about the distribution of the data are adopted. Semi-supervised classification (SSC) and semi-supervised regression (SSR) are the key components of SSL [5], depending upon the type of the output variable.

In SSL, there are several assumptions that define the forms of intended interaction [6]. The most widely adopted assumptions are as follows [4]:

- Smoothness assumption: two samples close to each other in the input space should result in close labels in the output space as well.
- Low-density assumption: the decision margin should not intersect across densely populated portions of the input space.
- Manifold assumption: the labels for sample points on the same low-dimensional manifold should be the same.

Most, if not all, SSL algorithms are built on one or more of these assumptions. These assumptions are different definitions of the similarity between data points and their patterns [4]. They guide how SSL algorithms perceive relationships within the dataset. For instance, the smoothness assumption suggests that neighboring data points in the input space should have similar labels, reflecting a continuous pattern within the data.

The two most prevalent divisions in SSL are transductive and inductive, which are founded on the purpose of the training process. The former is merely concerned with providing labels for unlabeled data (not providing a model), whereas the latter constructs a classification or regression model that can be used to predict the label of unseen data points.

As adaptations of preexisting supervised algorithms, inductive learning algorithms that are called intrinsically semi-supervised [4] allow unlabeled data to be included in the objective function. They do not use any intermediary stages or supervised base learners, rather they directly optimize an objective function with components for labelled and unlabeled data. In general, these algorithms rely on one of the SSL assumptions, either explicitly or implicitly, and most SS neural networks rely on the smoothness assumption.

In the present work, a SS deep neural network (SSDNN), by implicitly implementing the HIs' criteria as well as using the available EoL, has been proposed to construct HI by feature fusion. First, a hypothetical ideal HI kernel function following the HIs' criteria is proposed and then used as a target for a supervised ANN to approximate the HI function (see Figure 3.1). In this regard, two main questions arise:

1. How to select a suitable function to make an ideal HI conforming to the prognostic (HIs') criteria?

2. What variable should be considered as the main variable of the ideal HI function?

which are discussed in the following.

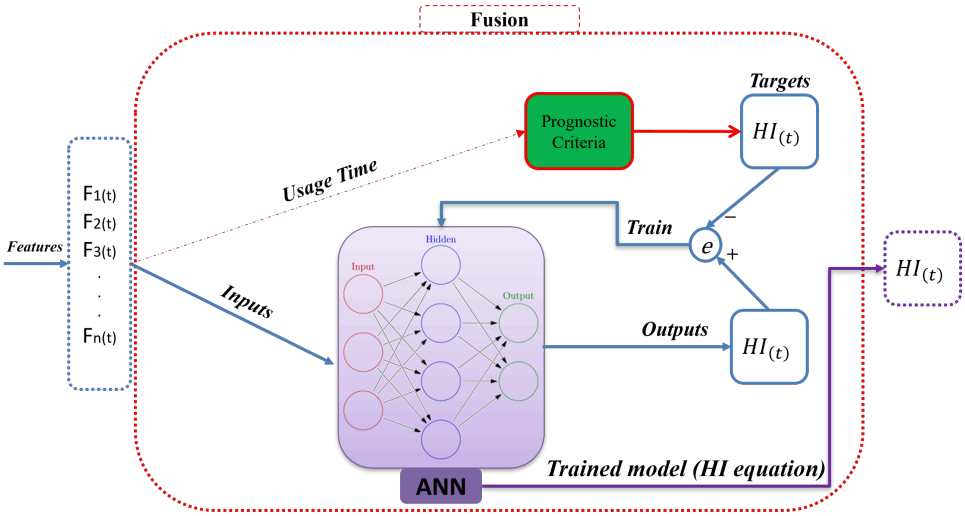


Figure 3.1: Semi-supervised criteria-based NN Fusion to construct the HI from the features.

Selecting hypothetical kernel function to simulate ideal HI

To create targets of the NN to predict HI, three aspects have been considered:

- The ideal HI function should best conform to all the criteria (Mo, Tr, and Pr).
- The simpler the ideal HI function, the simpler the NN model, and the faster its convergence.
- The ideal HI function should take into account the nonlinear behavior of damage propagation and accumulation in an engineering system or structure.

The smoothness assumption of SSL is already taken into account when using an ideal HI function as a label generator that fulfils the HIs' criteria, i.e., if two bunches of extracted features at two different time steps are close to each other in the input space, their HIs (labels) are close to each other as well. Inversely, increasing the dissimilarity between two groups of extracted features at two separate time steps in the input space causes their HIs to move apart. In other words, the relative RUL between the former $(t-1)$ and current (t) time windows, from which the features are extracted, is known. As a result, a direct relationship between the relative RUL (δRUL) and the relative HI variation (degradation) (δHI) can be used to reconstruct the relative HI. By using the last relative RUL at the last time window before EoL (which is the only labeled data point concerning RUL for each sample) and considering a threshold as the maximum HI at EoL, all HI labels are recursively provided from the final failure at the EoL to the healthy state at the onset, yielding simultaneously prognosable behavior in HIs as the labels (see Figure 3.2).

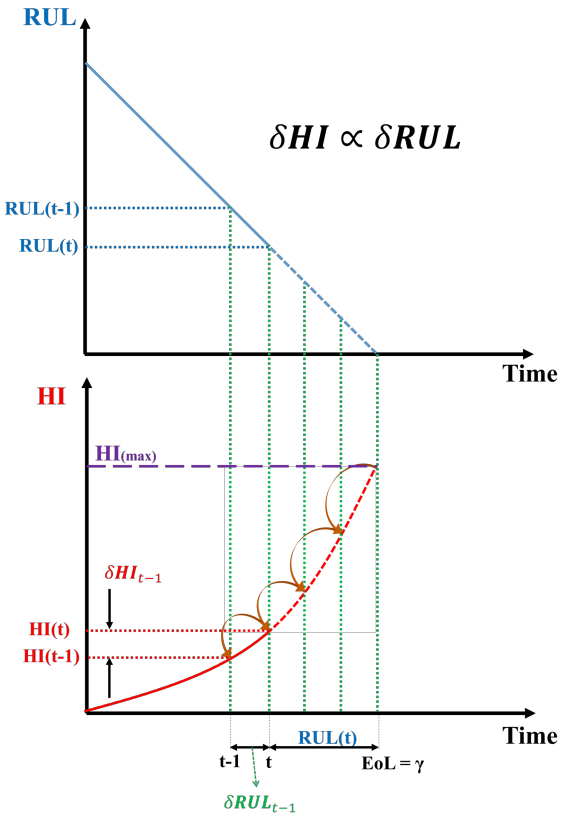


Figure 3.2: Recursive reconstruction of the HI labels based on the relative RUL (δRUL) and the maximum HI at EoL to implement SSL.

With the abovementioned hypotheses in mind, four basic kernel functions having high compatibility with the criteria are proposed and studied to select the best one: linear ($HI_t = t$), quadratic polynomial ($HI_t = t^2$), natural logarithm ($HI_t = \ln(t)$), and exponential functions ($HI_t = \exp(t)$). These functions should be expressed in terms of usage time, which in this case is fatigue cycles. The functions should be normalized using the max-min normalization to adapt Pr as a recursive reconstruction process of HI. This normalization process is acceptable for hypothetical targets. In order to investigate the HIs' criteria, three different artificial units with variable lifetimes of 7, 4, and 10 time units (time step is 0.05) are considered, as shown in Figure 3.3. All functions best match to Mo and Pr according to the calculated criteria shown in Figure 3.4, however only linear and quadratic polynomial functions have the highest value (1) of Tr. As a result, the quadratic polynomial kernel function is used to construct the targets since it takes into account damage propagation and accumulation nonlinearity. Therefore, the equation of ideal hypothetical

HI for generating the targets is:

$$HI_{(t)} = normalize\left(t^2\right) = \frac{t^2 - HI_{min}}{HI_{max} - HI_{min}} = \frac{t^2 - t_0^2}{t_{EoL}^2 - t_0^2} \xrightarrow{t_0=0} HI_{(t)} = \frac{t^2}{t_{EoL}^2} \tag{3.10}$$

where t_0 and t_{EoL} are the operational times in terms of cycles at the beginning and the EoL, respectively. HI_{min} and HI_{max} are the minimum and maximum values of HI. The key point is that t_{EoL} is not available before the final failure. Using the simulated labels, a loss function at the output layer can be defined.

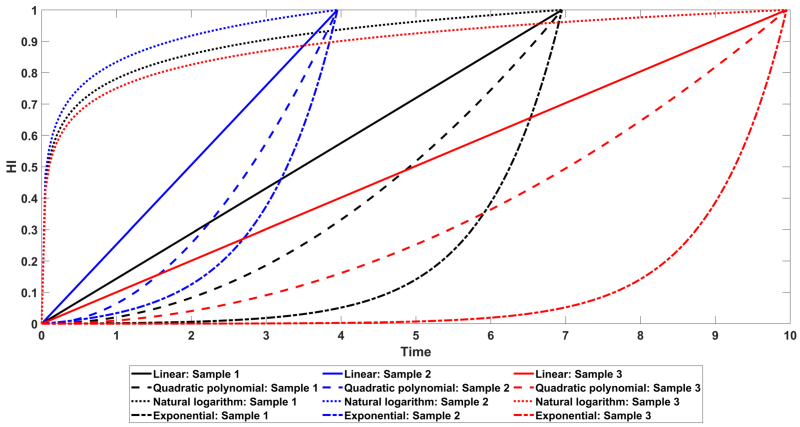


Figure 3.3: Hypothetical HI functions for three artificial units with different lifetimes.

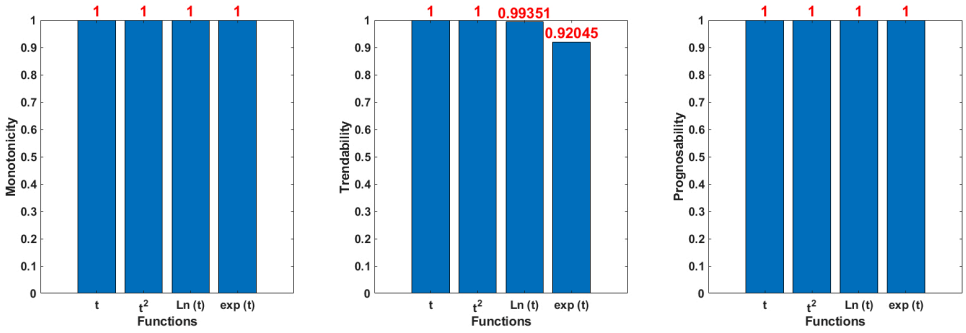


Figure 3.4: Prognostic criteria of four hypothetical HI functions shown in Figure 3.3.

3.4.1 Validation

The proposed SSL paradigm will serve as the core within newly developed frameworks, demonstrating its practicality in various case studies [1, 7–9]:

- **Chapter 5: Designing HI for T-Stiffener Composite Panels using Acoustic Emission:** implemented within three new frameworks.

- **Chapter 6: Designing HI for T-Stiffener Composite Panels and Dogbone Specimens using Guided Waves:** utilized within one new framework.

- **Chapter 7: Fusion of Acoustic Emission and Guided Waves Techniques:** integration of the new frameworks developed for AE and GW in Chapters 5 and 6 using SSL.

- **Appendix A.1: Interpretable HIs for Commercial Turbofan Engines:** implemented within one new framework containing the techniques developed for interpretability, elaborated in the next section.

These chapters collectively showcase the practical application and versatility of the proposed SSL paradigm across different applications, datasets, and frameworks.

3.5 Interpretability

In the pursuit of interpretable HIs, this section introduces a novel methodology with a focus on enhancing interpretability metrics, particularly the readability of the HI's equation—a direct correlation of its length. Recent studies have demonstrated the viability of constructing HIs through operations like addition and multiplication on derived features from sensory data [10]. The unique contribution here lies in the automatic integration of these mathematical operators within the ANN, facilitating the creation of efficient HIs while maintaining the inherent accuracy of deep learning. It is crucial to note that, rather than producing the equation as an output, the ANN inherently encapsulates the equation [9].

Diverging from conventional ANNs, this section first introduces a modified multiplicative neuron, marking a paradigm shift by incorporating multiplicative neurons alongside standard additive ones. The aim is to simplify the equation structure for a more inclusive and understandable model. This departure is exemplified through the design of HIs tailored for the commercial turbofan engine dataset (CMAPSS), showcasing the utilization of multiplication and division operators.

The significance of sparse weights is underscored, drawing inspiration from the acknowledgment that continuous weights for each neuron may lead to complex and non-understandable equations. Although an exploration ensues, introducing the development of a multiplicative neuron to complement conventional additive neurons, it becomes evident that this modification alone is insufficient for deriving a compact HI's equation from the ANN.

To address this limitation, a strategic approach is employed—discretizing continuous weights using a ternary set, coupled with sparsity control to guide convergence towards values -1, 0, 1 while controlling the number of zeroes. This results in an ANN comprising both additive and multiplicative neurons with discrete weights, aiming for simplicity and interpretability in the equation structure. The specific model designed based on these innovations will be investigated for the CMAPSS dataset in Appendix A.1, adding a practical dimension to these theoretical foundations.

3.5.1 Modified multiplicative layer

A limited number of neurons and layers should be employed in order to derive an effective equation that could characterize a HI. An ANN's compact size depends on the subject under examination. It is presumed that even a basic network of two 8-neuron layers could result in an outsized, physically unexplainable equation representing a HI. At first, it can seem extremely difficult for an ANN to be trained with just a few parameters and deliver correct results. A probable underfitting of the data is foreseen, even for small datasets. A straightforward HI equation can be derived from the ANN itself by including the physical parameters in it and zeroing out some weights in the training step. In the current work, it is regarded that physical properties could be basic multiplications and summations among features (like CMAPSS dataset [10]), which can be done by the combination of the multiplicative and additive layers, as will be seen in the next subsections. By discretizing the weights into a ternary shape and regulating the number of weights that should be zero, it is technically feasible to automatically decrease the number of neurons and further simplify the HI formula.

Introspecting ANN – additive neuron

Artificial neurons, which are coupled together and organized into layers, are the building blocks of an ANN. Each layer receives input from signals. One layer's output feeds into the subsequent layer's input. The basic equation of the typical ANN for each neuron individually given certain inputs x_K from the preceding layer is:

$$N_j = \sum_{i=1}^{K} [w_{ji}^{l} x_i] + b^l \tag{3.11}$$

where N_j is the initial output of neuron and w_{ji}^{l} is the weight relevant to the connection between the j^{th} neuron at the l^{th} layer to the $(l-1)^{th}$ layer's i^{th} neuron. The neuron also contains b^l to consider the bias. By using a nonlinearity through an activation function $F(N)$, which has the only restriction of being differentiable to the points of interest, the final output of the neuron is computed. While the ANN is being trained, the weights and biases of each neuron, which stand in for the learnable parameters of the network, are attempting to modify their values by backpropagating the error through the derivatives. A differentiable loss function must also be formulated at the points of interest. Due to the fact that this neuron sums the weighted inputs, it is termed additive.

Introspecting ANN – multiplicative neuron

The basic equation of the neuron Eq. 3.11 should be modified in order to induce the layers to generate multiplication operators. Accordingly, as stated in [11], we can construct a multiplicative neuron instead of a typical additive neuron by changing the summation step ($\sum_{i=1}^{K}[w_{ji}x_i]$) to a multiplication one ($\prod_{i=1}^{K}[x_i^{w_{ji}}]$), with the weights acting as exponents in a product rather than weights in a sum. Unfortunately, as mentioned in [12], an ANN with typical multiplicative layers makes the training more complex and slower due to the derivatives that are needed for backpropagation. This is the main reason that these layers have not been applied extensively in the literature. To mitigate this pitfall, a modified multiplicative neuron is developed by converting the additive neuron via a specific pair of

continuous activation functions. In particular, the inputs must get a logarithmic activation before being fed into Eq. 3.11 and an exponential activation afterward. The following equation can be used to update an additive neuron into a multiplicative one after the above-mentioned adjustments:

$$\begin{aligned} N_j &= e^{\sum_{i=1}^{K}[w_{ji}^{l} ln(x_i)]+b^l} \\ &= e^{b^l}.e^{\sum_{i=1}^{K}[ln(x_i)^{w_{ji}^{l}}]} \\ &= e^{b^l}.e^{ln(\prod_{i=1}^{K}[x_i^{w_{ji}^{l}}])} \\ &= e^{b^l}\prod_{i=1}^{K}[x_i^{w_{ji}^{l}}] \end{aligned} \tag{3.12}$$

The transition from additive to multiplicative neurons is shown in 3.5. An ANN can avoid adding further nonlinearities that might result in a complex equation by only employing these two types of activation functions. A key point to highlight is that by constraining the neurons to perform these particular activation functions, their ability to scale is confined by the requirement that the inputs be positive in order to apply the logarithm. Nonetheless, as the inputs could be simply rescaled to a desirable positive range, this is not a limitation in the current research. Furthermore, the convergence principles of neural networks are fulfilled given that the logarithm exists because the proposed multiplicative neuron derives naturally from the additive one by adjusting the activation functions.

3.5.2 Discretized weights

Learning the weights in continuous spaces is very favorable since the training process is stable and the optimal solution can be properly found. Nevertheless, because the ANN design is typically complex with vast numbers of weights, this is not effective in developing concise equations. This is particularly important in the case of HI construction, where the objective is to recognize the pattern and reconstruct a HI that offers high criteria scores (Mo, Tr, and Pr) rather than merely exact target values [1]. Incorporating continuous values with multiple decimal digits provides a complex model, even in extreme situations where only a small number of weights are non-zero. An ANN cannot be trained with discontinuous weights because there are no gradients for back-propagation; hence, learning in a continuous space is ultimately inevitable. Rounding the weights to the desired decimal during testing could be an easy way to achieve a compact formula, yet it would adversely affect the outputs and most likely result in the ANN being ineffective.

The weights should preferably be discrete to particular decimal values or even integers without compromising precision. Ternary weights have lately been developed to help with this challenge [13]. The objective is to train an ANN by converging the weights to specified values instead of rounding them to particular decimal digits. They are known as "ternary" values if the provided values are `{-1, 0, 1}`. There are undoubtedly scenarios where we require weights to fall within the range of those integers. This approach only induces a portion of the weights, which is controllable, to be integers rather than forcing all.

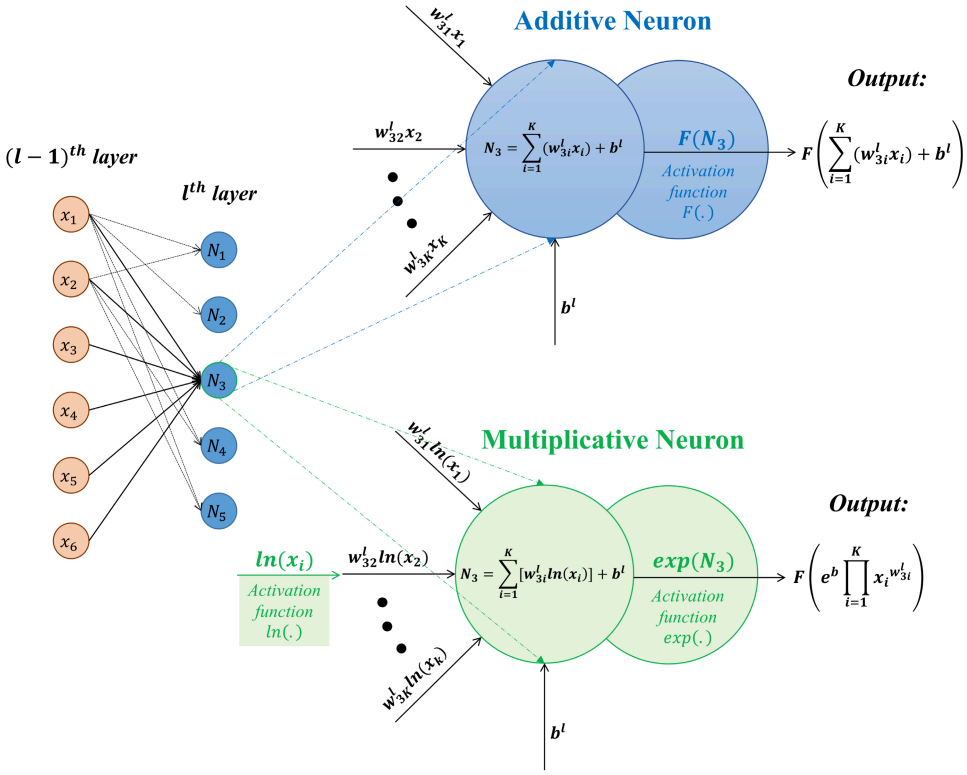


Figure 3.5: Additive and multiplicative neurons.

According to Ref. [13], the full-precision weight space is too vast to identify an acceptable ternary solution, thereby the continuous weight spaces need to be constrained by $tanh(w)$:

$$w' = tanh(w) \tag{3.13}$$

The weights are now bounded to the chosen [-1, 1], hyperbolic tangent range. Adding just one more term to the loss function (E) enables this transition work:

$$E = E_C(y, \hat{y}) + \lambda E_R(w') \tag{3.14}$$

$$E_C(y, \hat{y}) = \frac{1}{n}\sum_{j=1}^{n}(y_j - \hat{y}_j)^2 \tag{3.15}$$

$$E_R(w') = E_R(tanh(w)) = \sum_{l=1}^{L}\sum_{w_{ji}}^{|w^l|}[(\alpha - tanh^2(w^l_{ji}))tanh^2(w^l_{ji})] \tag{3.16}$$

where $E_C(y, \hat{y})$ is the mean-squared loss (MSE) between labels y_j and predicted outputs $\hat{y}_j$ over n data points (as an example loss function for the regression task), and $E_R(w')$ is

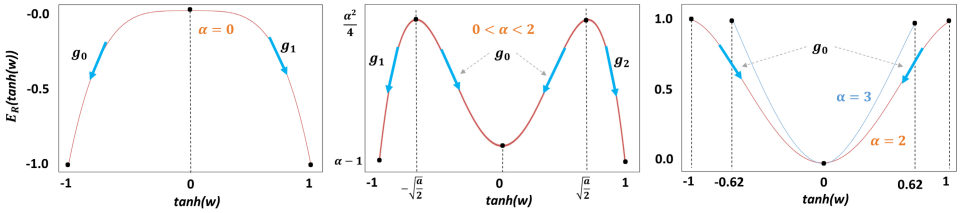


Figure 3.6: Gradients' flow of the discretizing loss function E_R during training for different shape controller parameters α. The desired local minima exist when $0 < \alpha < 2$.

the discretizing loss for converging the weights towards the ternary values. λ is a regularization parameter, L is the number of layers, $|w^l|$ denotes the total number of weights for the l^{th} layer (the summation is over all w_{ji} which is the weight relevant to the connection between the j^{th} neuron at the l^{th} layer to the $(l-1)^{th}$ layer's i^{th} neuron), and α is the shape controller of $E_R(.)$. λ and α are supplementary hyperparameters that should be adjusted for training the ANN; the first can be seen as a trade-off between the importance of reducing the MSE and ternarizing more weights effectively, and the latter softly controls the number of weights to become zero. The gradients exist and are proven to be minimal at $tanh(w) = -1$, $tanh(w) = 0$ and $tanh(w) = 1$ when $0 < \alpha < 2$ using the aforementioned modifications and loss functions (the proof in [13]). The $E_R(w')$ for different α values is shown in Figure 3.6. The number of zeros in the trained weights, which can be monitored to have more or fewer parameters, is the sparsity control key property of Eq. 3.16. This is especially helpful in situations where larger ANN architectures were obtained, yet we still want to have concise formulations for HIs by zeroing (raising α) more weights. By maintaining the weights as close to their ternary shape as appropriate and regulating the proportion of them that should be identical to zero, the ANN is able to generate precise predictions, which is a benefit of the adjustment to the weights and the incorporation of the term to the total loss function E.

3.5.3 Validation

The proposed techniques, namely the incorporation of multiplicative neurons and sparsity control, undergo rigorous testing using the NASA Ames Prognostics Data Repository dataset for turbofan engine degradation simulation (CMAPSS) in Appendix A.1. The developed model makes substantial contributions by introducing a novel type of neuron, constructing a network with a synergy of additive and multiplicative neurons, and harnessing the advantages of multiplicative neurons and sparsity control through discretized weights. The resulting HIs exhibit concise and easy-to-understand equations, all while satisfying essential evaluation criteria (Mo, Pr, and Tr). Appendix A.1 will delve into a comparative analysis with PCA, KPCA, and two-stage genetic programming (GP) outputs, shedding light on the effectiveness and interpretability of the proposed methodologies.

3.6 Conclusions

In conclusion, this chapter presented methodologies for the development and evaluation of HIs, emphasizing the necessity of a nonlinear approach aligned with the physics

of damage propagation and accumulation. Key criteria for HIs, including Mo, Tr, and Pr, were introduced as pivotal elements for assessing HIs' quality. A significant shift towards evaluating HIs specifically on test units, crucial for real-world applications, was highlighted to enhance generalizability.

The chapter introduced a novel SSL paradigm, leveraging HIs' criteria as supervisors across various applications and model architectures. Furthermore, the chapter delved into the development of interpretable models, providing valuable insights into the interpretability aspect of HIs. The emphasis was placed on the readability of the HI's equation, introducing a modified multiplicative layer alongside discretized weights to achieve concise and interpretable formulas for turbofan engines.

In summary, the chapter laid the foundation for a robust and versatile methodology for HI development, incorporating nonlinearities, emphasizing criteria-based evaluations, and introducing innovations in interpretability. Subsequent chapters are anticipated to validate and extend these methodologies through practical applications and comparative assessments.

References

[1] M. Moradi, A. Broer, J. Chiachío, R. Benedictus, T. H. Loutas, and D. Zarouchas, "Intelligent health indicator construction for prognostics of composite structures utilizing a semi-supervised deep neural network and shm data," *Engineering Applications of Artificial Intelligence*, vol. 117, p. 105502, 2023.

[2] J. Coble and J. W. Hines, "Identifying optimal prognostic parameters from data: a genetic algorithms approach," in *Annual Conference of the PHM Society*, vol. 1, 2009.

[3] N. Eleftheroglou, "Adaptive prognostics for remaining useful life of composite structures," 2020.

[4] J. E. Van Engelen and H. H. Hoos, "A survey on semi-supervised learning," *Machine learning*, vol. 109, no. 2, pp. 373–440, 2020.

[5] G. Kostopoulos, S. Karlos, S. Kotsiantis, and O. Ragos, "Semi-supervised regression: A recent review," *Journal of Intelligent & Fuzzy Systems*, vol. 35, no. 2, pp. 1483–1500, 2018.

[6] C. Olivier, "Semi-supervised learning (adaptive computation and machine learning)," *Mit Pr, 2006*, 2006.

[7] M. Moradi, A. Broer, J. Chiachío, R. Benedictus, and D. Zarouchas, "Intelligent health indicators based on semi-supervised learning utilizing acoustic emission data," in *European Workshop on Structural Health Monitoring*, pp. 419–428, Springer, 2022.

[8] M. Moradi, J. Chiachío, and D. Zarouchas, "Developing health indicators for composite structures based on a two-stage semi-supervised machine learning model using acoustic emission data," in *Proceedings of the 10th ECCOMAS Thematic Conference on Smart Structures and Materials*, vol. 10, 2023.

[9] M. Moradi, P. Komninos, R. Benedictus, and D. Zarouchas, "Interpretable neural network with limited weights for constructing simple and explainable hi using shm data," in *Annual Conference of the PHM Society*, vol. 14, 2022.

[10] K. T. Nguyen and K. Medjaher, "An automated health indicator construction methodology for prognostics based on multi-criteria optimization," *ISA transactions*, vol. 113, pp. 81–96, 2021.

[11] R. Durbin and D. E. Rumelhart, "Product units: A computationally powerful and biologically plausible extension to backpropagation networks," *Neural computation*, vol. 1, no. 1, pp. 133–142, 1989.

[12] M. Schmitt, "On the complexity of computing and learning with multiplicative neural networks," *Neural Computation*, vol. 14, no. 2, pp. 241–301, 2002.

[13] X. Deng and Z. Zhang, "Sparsity-control ternary weight networks," *Neural Networks*, vol. 145, pp. 221–232, 2022.

4

Data Analysis and Machine Learning

In this foundational chapter, key operations essential for subsequent frameworks in the thesis are introduced, covering preprocessing techniques, signal processing methods, statistical features, dimension reduction, and an overview of deep learning. They equip readers with essential knowledge for designing frameworks across diverse applications and datasets in the following chapters.

4.1 Introduction

In this chapter, the groundwork for data analysis is laid, introducing essential operations pivotal for subsequent frameworks developed in this thesis. The chapter begins with fundamental preprocessing techniques, encompassing normalization, standardization, and a newly developed adaptive standardization. Following this, signal processing methods such as fast Fourier transform (FFT), Hilbert transform (HT), and complete ensemble empirical mode decomposition with adaptive noise (CEEMDAN) are explored to cater to diverse data handling needs. Statistical features, offering insights into temporal and spectral characteristics in both time and frequency domains, are outlined. Additionally, dimension reduction through principal component analysis (PCA)-based algorithms for simplifying complex datasets is presented. The chapter concludes with an overview of deep learning, exploring architectures like multilayer perceptron (MLP), convolutional neural network (CNN), long short-term memory (LSTM), and bidirectional LSTM (BiLSTM), along with insights into hyperparameter tuning and ensemble learning (EL) strategies. Serving as an instructive guide, this chapter equips readers with the essential understanding of data analysis techniques crucially employed in designing new frameworks across various applications and datasets in the subsequent chapters.

4.2 Preprocessing

Data transformation involves transforming the data into formats that are suitable for mining. Normalization is a primary category of transformation in which the attribute data is scaled to fall into a narrow range, like -1.0 to 1.0 or 0.0 to 1.0 [1, 2]. Algorithms benefit from normalization, especially neural networks. Normalizing input values in training sets for every attribute speeds up the learning process in classification mining using the backpropagation algorithm. It also contributes to distance-based techniques by keeping attributes with broad ranges from outweighing those with narrower ranges [1].

Normalization (referring to min-max normalization) or standardization (referring to zero-mean normalization) are commonly applied preprocessing techniques for inputs before being fed into a model. These advantageous preprocessing techniques are also employed between the hidden layers of an ANN model. However, they should be utilized with caution in the field of prognostics or other fields with a similar objective because future data is not accessible to the model, especially during its testing phase. In fact, whether the input data are normalized or standardized given the entire trajectory, the input data distribution is already placed into a known distribution. In this manner, the problem has somehow been turned into an interpolation task for the data-driven model when it is in fact an extrapolation task, which is more difficult for ML models. This point is crucial for assessing prognostic-relevant methods, such as HI construction or RUL prediction models. As a result, an adaptive standardization technique needs to be developed. After outlining the normalization and standardization relationships, this section will introduce the adaptive standardization technique.

4.2.1 Normalization (min-max)

The original data is transformed linearly by min-max normalization:

$$x_i^{norm} = \frac{x_i - min(X)}{max(X) - min(X)} \quad \text{where} \quad X = x_{1:N} \tag{4.1}$$

where x_i is the i^{th} data at the i^{th} time step (t_i), and i basically refers to the current (data or time step). X indicates the entire historical data from the beginning to the end N (EoL).

4.2.2 Standardization (zero-mean)

The value of an input feature is standardized using the mean μ and standard deviation σ of all the data for that feature when using zero-mean normalization, also known as z-score normalization:

$$x_i^{st} = \frac{x_i - \mu}{\sigma} \quad \text{where} \quad \mu = \frac{\sum_{j=1}^{N} x_j}{N} \; ; \; \sigma = \sqrt{\frac{\sum_{j=1}^{N} \left(x_j - \mu\right)^2}{N}} \tag{4.2}$$

When the input data has a Guassian distribution, this method performs better.

4.2.3 Adaptive standardization

According to the explanations above, the plausible standardization technique can be used for both training and test data based on the mean value and standard deviation derived exclusively from the training data [3]. However, because the same technique was employed and noticed its misleading impacts on results, this may not be useful for new unseen data (validation or test portions). As a result, an adaptive standardizing technique is being developed. Assuming that μ_i and σ_i are the mean value and standard deviation of the data ($x_{1:i}$) up to the present (time step t_i), the data are standardized as follows:

$$x_i^{ast} = \frac{x_i - \frac{\sum_{j=1}^{i-1} x_j^{ast} + x_i}{i}}{\sqrt{\frac{\sum_{j=1}^{i-1}\left(x_j^{ast} - \frac{\sum_{j=1}^{i-1} x_j^{ast} + x_i}{i}\right)^2 + \left(x_i - \frac{\sum_{j=1}^{i-1} x_j^{ast} + x_i}{i}\right)}{i}}} = \frac{x_i - \mu_i}{\sigma_i} \tag{4.3}$$

$$\mu_i = \frac{\sum_{j=1}^{i-1} x_j^{ast} + x_i}{i} \tag{4.4}$$

$$\sigma_i = \sqrt{\frac{\sum_{j=1}^{i-1}\left(x_j^{ast} - \mu_i\right)^2 + \left(x_i - \mu_i\right)}{i}} \tag{4.5}$$

It is important to note that the data preceding the current time step ($x_{1:i-1}$) is not standardized at the current time step (t_i), but has been standardized in previous time steps ($t_1, t_2, ..., t_{i-1}$). Only the current data (x_i) undergoes the standardization process at the current time step (t_i), rather than the preceding data. This procedure is carried out for the extracted features of each unit independently, which is acceptable and applicable from the prognostics standpoint.

4.3 Signal Processing

A signal, conveying essential information, is a crucial element in various applications. As signals acquired during tests often contain noise and bootless components, "signal processing" (SP) methods become imperative. SP aids in extracting system fault-related information, enhancing the signal-to-noise ratio (SNR). It has risen to the top of the manufacturing and operation of modern equipment, accounting for 50–70% of the total cost of the facility [10].

Three domains, namely time, frequency, and time-frequency, categorize SP methods. Three fundamental SP methods—fast Fourier transform (FFT), Hilbert transform (HT), and complete ensemble empirical mode decomposition with adaptive noise (CEEMDAN)—are very helpful to process data. These methods will be employed in subsequent chapters within newly developed frameworks for diverse applications. FFT efficiently transforms signals from the time domain to the frequency domain, offering insights into various frequency components. The HT is instrumental in analyzing instantaneous frequency and phase. CEEMDAN, being versatile, allows adaptive decomposition of signals into intrinsic mode functions, facilitating a detailed exploration of signal components. The utilization of these methods within the upcoming frameworks aims to enhance information extraction from signals across a range of scenarios. Thus, FFT, HT, and CEEMDAN will be briefly introduced in the next subsections.

4.3.1 FFT

FFT is a potent signal processing method for effectively analyzing signals in the frequency domain [4]. By converting a signal from the time domain to the frequency domain, the signal's frequency components are perceived. Due to its faster computational speed than other methods, the discrete Fourier transform (DFT) can be computed using the FFT algorithm, which is highly valuable.

The mathematical representation of the FFT can be expressed as follows:

$$X(k) = \frac{1}{N}\sum_{n=0}^{N-1} x(n)\exp\left(-\frac{j2\pi kn}{N}\right) \tag{4.6}$$

where, $x(n)$ indicates the discrete input signal at the time index n, and $X(k)$ represents the corresponding frequency component at the frequency index k. The exponential term, which involves complex numbers, contributes in capturing the signal's phase and amplitude information in the frequency domain.

FFT is helpful in many applications, including image analysis, communications, and audio signal processing, because of its capacity to effectively explore the spectral characteristics of signals. Its use in the frameworks developed in the upcoming chapters highlights how crucial it is to extracting valuable features from input data.

4.3.2 HT

HT for a signal $x(t)$, which denotes $\hat{x}(t)$, is expressed as [5]:

$$\hat{x}(t) = H[x(t)] = \frac{\int_{-\infty}^{\infty} \frac{x(t)}{t-\tau} d\tau}{\pi} = x(t) * \frac{1}{\pi t} \tag{4.7}$$

In Eq. 4.7, the symbol of $*$ depicts the convolution operation. By assuming:

$$j = \sqrt{-1}\,; \qquad a(t) = \sqrt{x(t)^2 + \hat{x}(t)^2}; \qquad \theta(t) = arctan\,(\frac{\hat{x}(t)}{x(t)}) \tag{4.8}$$

the analytic signal can be expressed as:

$$z(t) = x(t) + j\hat{x}(t) = a(t)\,exp[j\theta(t)] \tag{4.9}$$

where instantaneous amplitude and phase have been depicted by $a(t)$ and $\theta(t)$, respectively. In addition, the instantaneous frequency of the signal $x(t)$ can be obtained from $\omega(t) = \frac{d\theta(t)}{dt}$. The Hilbert envelope spectrum is also defined as:

$$h(f) = \int_{-\infty}^{\infty} a(t)\exp(-j2\pi ft)dt = \int_{-\infty}^{\infty} \left\{ \sqrt{x(t)^2 + \left[\frac{\int_{-\infty}^{\infty} \frac{x(t)}{t-\tau} d\tau}{\pi} \right]^2} \exp(-j2\pi ft) \right\} dt \tag{4.10}$$

The vibration signals obtained by a monitoring system are usually modulated when a mechanical fault happens. To demodulate a signal, HT is a popular technique to distinguish carrier component and modulation component in which the fault features are mostly concealed. It is particularly a useful SP method to process GW signals for damage detection and monitoring in composite structures [6].

4.3.3 CEEMDAN

Empirical mode decomposition (EMD) is an algorithm capable of decomposing non-linear, non-stationary signals into a set of orthogonal components. The basis of EMD is the Hilbert-Huang transform [7], which decomposes the original signal into simple intrinsic mode functions called IMFs. Each IMF needs to fulfill two conditions:

1. The number of extrema and zero crossings must be equal or differ by no more than one.
2. The mean value of the upper and lower envelopes is zero everywhere.

The basic steps of the EMD algorithms are explained in the following.

First, the local extrema of the data are identified. By employing cubic splines, an upper and lower signal envelope are created, containing the entirety of the signal data. Then, the mean of the upper and lower envelopes is calculated. In the next step, the difference between the mean (m_1) and the data ($X(t)$) is calculated, which corresponds to the first IMF component h_1:

$$X(t) - m_1 = h_1 \tag{4.11}$$

In case the two conditions mentioned above are not validated, the previous steps are repeated until h_1 complies with those criteria (sifting process). h_1 now corresponds to the new data, i.e., $X(t) = h_1$, and the previous steps are repeated to extract the next IMF as:

$$h_1 - m_{11} = h_{11} \tag{4.12}$$

After k siftings, the IMF that adheres to the criteria is designated as $c_1 = h_{1k}$. Then, c_1 is the first IMF and is subtracted from the original data:

$$X(t) - c_1 = r_1 \tag{4.13}$$

where r_1 is the first residual and is treated as the new data in Eq. 4.11. The subsequent steps are to extract the n^{th} IMF (c_n) until r_n is a monotonic function. When a monotonic function is reached, all IMFs are extracted, and the process is completed. To reconstruct the original data after extracting the n^{th} IMF and leaving the residual r_n, Eq. 4.15 is used:

$$X(t) = \sum_{i=1}^{n} c_i + r_n \tag{4.14}$$

Despite the advantages EMD provides in signal decomposition, it suffers from a major drawback referred to as mode mixing. A solution to this drawback is ensemble EMD (EEMD) [8], which adds Gaussian white noise to the data with the appropriate scale. The steps of EEMD are the same as regular EMD, with the addition of white Gaussian noise to the original data. Then the extracted IMFs are averaged. However, the addition of white noise creates new issues, such as the independence of the decomposition process due to the residual noise in the IMF and deficiencies in the decomposition process. Also, the EEMD makes it difficult to average different numbers of IMFs. These newfound issues are addressed by the complete ensemble empirical mode decomposition with adaptive noise (CEEMDAN) [9]. The CEEMDAN process, unlike the EEMD, adds particular noise $E_j[w_i(t)]$ at each step of the decomposition. The main process of CEEMDAN is described in the following steps:

$$x_i(t) = X(t) + \beta_0 E_1[w_i(t)] \tag{4.15}$$

where $E_1[w_i(t)]$ is added to the original signal $X(t)$ for $i = 1, 2, \ldots, N$. The parameters w_i, β, and N indicate the added Gaussian noise, the noise amplitude, and the ensemble size, respectively. The first IMF (c_1) is calculated through the first residual r_1 as:

$$c_1 = X(t) - r_1, \quad where \quad r_1 = \frac{1}{N}\sum_{i=1}^{N} M(x_i(t)) \tag{4.16}$$

where $M(\bullet)$ is the operator representing local means of data. The second IMF (c_2) is obtained through Eq. 4.17:

$$c_2 = r_1 - r_2, \quad where \quad r_2 = \frac{1}{N}\sum_{i=1}^{N} M\left(r_1 + \beta_1 E_2[w_i(t)]\right) \tag{4.17}$$

Here $E_2[w_i(t)]$ is the second IMF of EEMD. The n^{th} IMF of CEEMDAN is obtained through:

$$c_n = r_{n-1} - r_n, \quad where \quad r_n = \frac{1}{N}\sum_{i=1}^{N} M\left(r_{n-1} + \beta_{n-1} E_n[w_i(t)]\right) \tag{4.18}$$

where $\beta_n = \varepsilon_0 std(r_n)$ is the signal-to-noise ratio (SNR).

4.4 Statistical Features

In this step, features are extracted from multiple domains, including time and frequency domain. FE can also be carried out based on physical models which imply physical meaning but these model-based features are limited. As a result, statistical parameters are extracted as features from various domains and employed in the following step (fusion).

The popular features in the time domain have been listed in Table 4.1. However, as previously explained, the time domain is not sufficient to extract features as a HI or an element of HI, and additional statistical features should be extracted from the frequency domain. Furthermore, since an incomprehensible variation, especially in high frequency fluctuations, may not be detected in the time domain and instead it simply causes a spectrum line in the frequency domain, the frequency spectrum is more susceptible to incipient damages. This case is widely used in fault detection. The common features in the frequency domain have been listed in Table 4.2.

Table 4.1: Common statistical features in time domain.

No	Equation	Specific name	No	Equation	Specific name
1	$X_m = \frac{\sum_{n=1}^{N} x(n)}{N}$	Mean Value	9	$X_{crest} = \frac{X_{peak}}{X_{rms}}$	Crest factor [60]
2	$X_{sd} = \sqrt{\frac{\sum_{n=1}^{N}(x(n)-X_m)^2}{N-1}}$	Standard Deviation	10	$X_{clearance} = \frac{X_{peak}}{X_{root}}$	Clearance factor
3	$X_{root} = \left(\frac{\sum_{n=1}^{N}\sqrt{\lvert x(n)\rvert}}{N}\right)^2$	Root Amplitude	11	$X_{shape} = \frac{X_{rms}}{\frac{1}{N}\sum_{n=1}^{N}\lvert x(n)\rvert}$	Shape factor
4	$X_{rms} = \sqrt{\frac{\sum_{n=1}^{N}(x(n))^2}{N}}$	Root Mean Square (RMS) [61]	12	$X_{impulse} = \frac{X_{peak}}{\frac{1}{N}\sum_{n=1}^{N}\lvert x(n)\rvert}$	Impulse factor
5	$X_{rss} = \sqrt{\sum_{n=1}^{N}\lvert x(n)\rvert^2}$	Root-sum-of-squares (RSS) [61] (The RSS level is also referred to as the 2-norm.)	13	$X_{p2p} = \max(x(n)) - \min(x(n))$	Maximum to Minimum difference
6	$X_{peak} = max\lvert x(n)\rvert$	Peak	14, 15, 16, 17	$X_{k_c m} = \frac{\sum_{n=1}^{N}(x(n)-X_m)^k}{N}$	Central moment for k^{th} order (In this context, the 3^{rd}, 4^{th}, 5^{th}, and 6^{th} are considered the features of 14, 15, 16, and 17, respectively.)
7	$X_{skewness} = \frac{\sum_{n=1}^{N}(x(n)-X_m)^3}{(N-1)X_{sd}^3}$	Skewness	18	$X_{FM4} = \frac{X_{4\_cm}}{X_{sd}^4}$	FM4 (close to Kurtosis)
8	$X_{kurtosis} = \frac{\sum_{n=1}^{N}(x(n)-X_m)^4}{(N-1)X_{sd}^4}$	Kurtosis	19	$X_{med} = \frac{\sum_{n=1}^{N} t(n)}{N}$	Median

$x(n)$, N, and $t(n)$ denote the signal sequence for $n = 1, 2, \ldots, N$, the number of data points, and time instances when $x(n)$ occurs.

Table 4.2: Common statistical features in frequency domain.

No	Equation	Specific name	No	Equation	Specific name
20	$p_1 = X_{mf} = \frac{\sum_{k=1}^{K} s(k)}{K}$	Mean Frequency (may indicate the vibration energy in the frequency domain, which represents the average of the amplitudes of all the frequencies.)	27	$p_8 = \sqrt{\frac{\sum_{k=1}^{K} f_k^4 s(k)}{\sum_{k=1}^{K} f_k^2 s(k)}}$	
21	$p_2 = \frac{\sum_{k=1}^{K}(s(k)-p_1)^2}{K-1}$	(same as variance)	28	$p_9 = \frac{\sum_{k=1}^{K} f_k^2 s(k)}{\sqrt{\sum_{k=1}^{K} s(k)\sum_{k=1}^{K} f_k^4 s(k)}}$	
22	$p_3 = \frac{\sum_{k=1}^{K}(s(k)-p_1)^3}{K(\sqrt{p_2})^3}$	(same as Skewness)	29	$p_{10} = \frac{p_6}{p_5}$	
23	$p_4 = \frac{\sum_{k=1}^{K}(s(k)-p_1)^4}{Kp_2^2}$	(similar to Kurtosis)	30	$p_{11} = \frac{\sum_{k=1}^{K}(f_k-p_5)^3 s(k)}{Kp_6^3}$	
24	$p_5 = X_{fc} = \frac{\sum_{k=1}^{K} f_k s(k)}{\sum_{k=1}^{K} s(k)}$		31	$p_{12} = \frac{\sum_{k=1}^{K}(f_k-p_5)^4 s(k)}{Kp_6^4}$	
25	$p_6 = \sqrt{\frac{\sum_{k=1}^{K}(f_k-p_5)^2 s(k)}{K}}$		32	$p_{13} = \frac{\sum_{k=1}^{K}\sqrt{(f_k-p_5)}s(k)}{K\sqrt{p_6}}$	
26	$p_7 = X_{rmsf} = \sqrt{\frac{\sum_{k=1}^{K} f_k^2 s(k)}{\sum_{k=1}^{K} s(k)}}$		33	$p_{14} = \sqrt{\frac{\sum_{k=1}^{K}(f_k-p_5)^2 s(k)}{\sum_{k=1}^{K} s(k)}}$	

$s(k)$, K, and f_k denote the spectrum for $n = 1, 2, \ldots, K$, the number of spectrum lines, and the frequency value of the k^{th} spectrum line.

4.5 Dimension Reduction – PCA-based Algorithms

Due to the large number of features obtained from SHM techniques, like AE data, the subsequent models become more complex. To address this issue, it is necessary to reduce the feature dimensionality. To accomplish this, PCA-based methods can be employed [10–13]. The aim is to decrease the number of features from more than a hundred to a few (e.g., 201 statistical features to 10 principal components (PCs) for AE data in Chapter 5). While additional PCs could be extracted and used as inputs for subsequent models, it is deemed acceptable to retain the reconstructed variance (90%) of features in order to maintain simplicity in the following models.

The application of PCA in the context of prognostics poses a challenge, as it is essential to avoid utilizing future data during the decomposition of eigenvectors (coefficients), especially when testing the model. In this thesis, two primary approaches are explored to applying PCA:

1. **Approach A:** The coefficients of PCs are extracted solely from the training units. Subsequently, these coefficients are utilized to construct PCs for the test units. There are two variations in this approach: A.1, where all data from the training units is concatenated (making a bigger matrix), and the coefficients are then extracted; and A.2, where coefficients are individually extracted from each training unit, and the mean value of the coefficients is used for testing.

   Approach B: Coefficients are independently extracted for each unit, regardless of whether they are from the training or test units. However, this approach is subject to two limitations. Firstly, the number of time steps during which measurements are collected must exceed the required number of PCs, due to matrix size requirements. Secondly, when considering the overall prognostics problem for each unit individually, the usage of future data (beyond the current time step) is prohibited, particularly during the test phase, due to the unavailability of future SHM measurements in practical scenarios.

Based on the authors' knowledge, approach B of estimating PCs yields higher performance when all data from the beginning until the EoL is considered. However, as mentioned earlier, the use of future data is impossible for prognostics. To address TD data, several extensions to PCA methods have been introduced, including dynamic PCA (DPCA), recursive PCA (RPCA), and moving-window PCA (MWPCA) [14]. While DPCA was developed to handle autocorrelation, RPCA and MWPCA are more suitable for dealing with nonstationary data. Among them, MWPCA forgets the older data (before window) entirely, which is not appropriate for the current prognostics purpose, especially when using passive SHM techniques like acoustic emission. To overcome this limitation, RPCA can be employed to incorporate all historical SHM data from the beginning of monitoring up to the current time. The approach B with all historical data till EoL is denoted as code B.1 (which is not acceptable in prognostics) and RPCA is coded as B.2, which is described in more detail in the following. As the dimension reduction stage inside the suggested framework, all approaches A.1, A.2, B.1, and B.2 are used and compared in one framework developed based acoustic emission data (Chapter 5). Although B.1 is not feasible in real-world applications, it is still used to demonstrate false positive performance and compare it with others.

In RPCA, assuming that the mean and covariance of all adaptively standardized observations $x_{1:i}^{ast}$ up to the present (time step t_i) are $\hat{\mu}_i$ and S_i, respectively, the updated mean $\hat{\mu}_{i+1}$ and the updated covariance matrix S_{i+1} given the new adaptively standardized observation x_{i+1}^{ast} (at time step t_{i+1}) can be calculated by:

$$\hat{\mu}_{i+1} = \left(1 - \frac{i}{i+1}\eta\right) x_{i+1}^{ast} + \frac{i}{i+1}\eta\hat{\mu}_i \tag{4.19}$$

$$S_{i+1} = \left(1 - \frac{i}{i+1}\eta\right)\left(x_{i+1}^{ast} - \hat{\mu}_{i+1}\right)\left(x_{i+1}^{ast} - \hat{\mu}_{i+1}\right)^T + \frac{i}{i+1}\eta S_i \tag{4.20}$$

where $0 \leq \eta \leq 1$ is a forgetting factor. In the context of a geometric progression, this is comparable to calculating a weighted mean and covariance of $x_{1:i+1}^{ast}$, where earlier values are down weighted exponentially. Older observations in RPCA are faster forgotten when using a forgetting factor of $\eta < 1$, whereas they are slower forgotten by applying a forgetting factor of $\eta \longrightarrow 1$. If it is desired to maintain all prior observations, which is appropriate for the current study, $\eta = 1$ should be adopted. By adopting RPCA, the proposed approach retains adaptability and compatibility with new incoming data while still maintaining the integrity of historical information critical for prognostics applications.

4.6 Deep Learning

While the primary objective of this research is not to identify the best and most universally optimal ANN's architecture for fusing SHM data, it is essential to provide and develop a suitable ANN architecture to showcase the feasibility of the suggested SSL paradigm. Given the novelty of the dataset for the examined composite structures and the absence of prior studies based on DL, various types of layers and architectures were explored. The approach began with basic shallow architectures like the multi-layer perceptron (MLP) to construct HI before progressing to more complex networks. Each layer was incrementally added, adjusting the number of neurons and introducing different layer types, including fully connected (FC) or dense layers, long short-term memory (LSTM), bidirectional LSTM (BiLSTM), and convolutional layers to enhance results. Considering diverse datasets, such as acoustic emissions (AE) and guided waves (GW), requires testing various layers and architectures. This section introduces key standard elements for the subsequent chapters, including FC, dropout, rectified linear units (ReLU), LSTM, BiLSTM, and CNN layers in the section on deep learning architectures and components. Additionally, a brief overview of hyperparameter tuning using the Bayesian optimization (BO) algorithm, frequently employed in the developed frameworks in the following chapters, is provided. Finally, ensemble learning (EL) methods, beneficial for reducing randomness and improving the performance of DNN models, are presented.

4.6.1 Deep learning architectures and components

Fully connected layer - FC

In FC layers, the neuron uses a weights matrix to apply a linear transformation to the input vector, which is called generalized linear layer ($z_j^l = \sum_k w_{jk}^l a_k^{l-1} + b_j^l$). A non-linear activation function σ is then used to apply a non-linear transformation to the product according to Eq. 4.21. If w_{jk}^l can be considered the weight for the link between the $l-1^{th}$

layer's k^{th} neuron to the l^{th} layer's j^{th} neuron, for a FC layer we have:

$$a_j^l = \sigma\left(z_j^l\right) = \sigma\left(\sum_k w_{jk}^l a_k^{l-1} + b_j^l\right) \tag{4.21}$$

where z_j^l and a_j^l denotes input and output of a desired FC layer l, respectively.

Dropout

Dropout is a regularization strategy that prevents complicated co-adaptations on training data, thereby decreasing overfitting in ANNs [15]. The following equation is considered for the dropout layer:

$$\hat{w}_j = \begin{cases} 0 & : with\ P(c) \\ w_j & : otherwise \end{cases} \tag{4.22}$$

where $\hat{w}_j$ is the diluted row and $P(c)$ is the probability c to remove a row in the weight matrix.

Rectified linear unit - ReLU

ReLU is a type of activation function with a strong biological and mathematical foundation [16]. It consists of setting a threshold at 0:

$$ReLU(z) = z^+ = max(0, z) \tag{4.23}$$

Long short-term memory - LSTM

A memory cell (g), an input gate (i), an output gate (o), and a forget gate (f) compose an LSTM unit, which was developed to maintain a long-term record of sequential inputs by using the memory unit [17]. As illustrated in Figure 4.1, one input (x_t) and the previous hidden state (h_{t-1}) as well as the previous cell state (c_{t-1}) are used to formulate the hidden state (h_t) in the t^{th} step in retaining information from the past as follows:

$$f_t = \sigma(W_f x_t + U_f h_{t-1} + b_f) \tag{4.24}$$

$$i_t = \sigma(W_i x_t + U_i h_{t-1} + b_i) \tag{4.25}$$

$$g_t = tanh(W_g x_t + U_g h_{t-1} + b_g) \tag{4.26}$$

$$o_t = \sigma(W_o x_t + U_o h_{t-1} + b_o) \tag{4.27}$$

$$c_t = f_t \odot c_{t-1} + i_t \odot g_t \tag{4.28}$$

$$h_t = o_t \odot tanh(c_t) \tag{4.29}$$

where W_* and b_* stand for learnable weights and bias parameters, respectively. σ is sigmoid activation function and $\odot$ is the element-wise product. The hidden state h_t influences the production of the final output at any step t by accumulating information from previously processed features [18], which could be referred to as damage accumulation and health degradation in the current study.

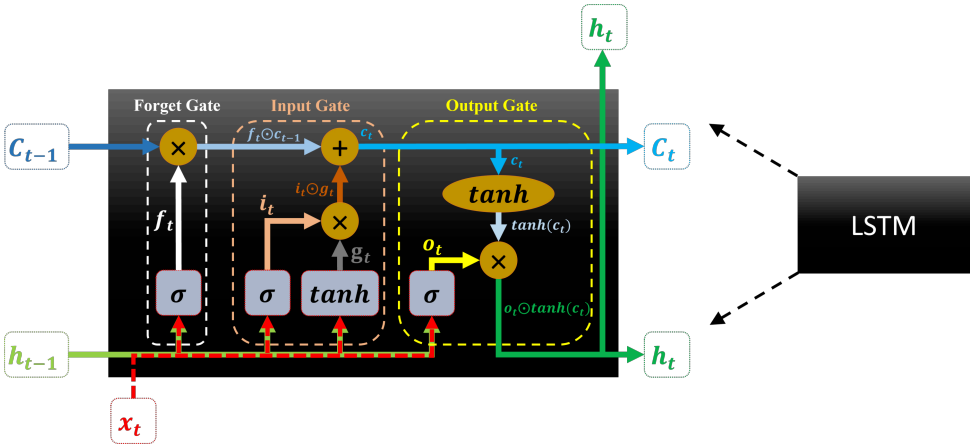


Figure 4.1: LSTM cell's architecture.

4

Bidirectional long short-term memory - BiLSTM

BiLSTM, a variant of recurrent neural networks (RNNs) primarily employed in natural language processing (NLP), stands out for its bidirectional information flow. Unlike standard LSTM, BiLSTM processes input sequences in both directions, allowing it to capture sequential dependencies from both ends [19]. This bidirectional capability is especially advantageous for modeling intricate relationships within sequences, such as words and phrases in NLP, and can be inspired to uncover hidden degradation patterns in SHM data.

In essence, BiLSTM introduces an additional LSTM layer that reverses the direction of information flow, as depicted in Figure 4.2. This implies that the input sequence is processed backward in this extra layer. The outputs from both LSTM layers are then combined through operations like average, sum, multiplication, or concatenation. It is important to note that, despite its advantages, BiLSTM comes with drawbacks, notably being a slower model that requires more training time compared to LSTM. Therefore, it is advisable to opt for BiLSTM only when its bidirectional capabilities are essential for the specific task at hand.

If we denote Eqs. 4.24-4.29 as representing forward LSTM operations, the corresponding equations for backward LSTM operations are:

$$f'_t = \sigma(W'_f x_t + U'_f h'_{t-1} + b'_f) \tag{4.30}$$

$$i'_t = \sigma(W'_i x_t + U'_i h'_{t-1} + b'_i) \tag{4.31}$$

$$g'_t = \tanh(W'_g x_t + U'_g h'_{t-1} + b'_g) \tag{4.32}$$

$$o'_t = \sigma(W'_o x_t + U'_o h'_{t-1} + b'_o) \tag{4.33}$$

$$c'_t = f'_t \odot c'_{t-1} + i'_t \odot g'_t \tag{4.34}$$

$$h'_t = o'_t \odot \tanh(c'_t) \tag{4.35}$$

BiLSTM combines the outcomes of both forward and backward LSTMs, enhancing the representation of the input sequence by incorporating insights from both past and future contexts.

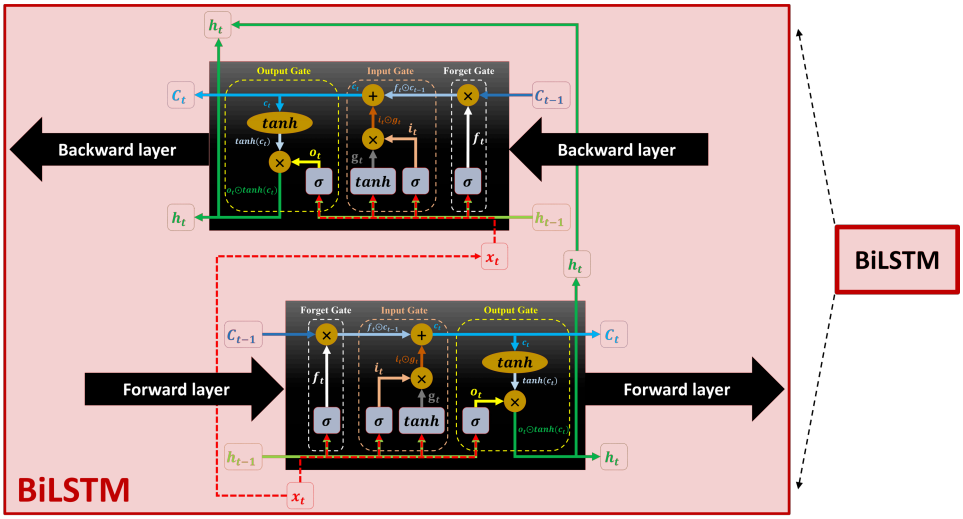


Figure 4.2: BiLSTM cell's architecture.

Convolutional neural network - CNN

A CNN is a type of neural network architecture designed for tasks involving grid-like data, such as images [20]. The basic components of a CNN include convolutional layers, pooling layers, and fully connected layers. For a 2D-CNN, considering a^{l-1} as the input coming from layer $l-1$, the equation is as follows:

$$Z_{ij}^{l} = f\left(\sum_{m=1}^{F_l}\sum_{n=1}^{K}\sum_{p=1}^{K} W_{mnp}^{l} a_{(i+m-1)(j+p-1)}^{l-1} + b_{ij}^{l}\right) \tag{4.36}$$

Here,

- Z_{ij}^{l} is the activation at position (i, j) in layer l,
- F_l is the number of filters in layer l,
- W_{mnp}^{l} is the weight of the connection between the m^{th} filter at position (n, p) in layer l and the input from layer $l-1$,
- $a_{(i+m-1)(j+p-1)}^{l-1}$ is the activation from position $(i+m-1, j+p-1)$ in layer $l-1$,
- b_{ij}^{l} is the bias term for the activation at position (i, j), and
- $f(\bullet)$ is the activation function.

In the following of the convolution layer, a pooling layer can be applied, like the 2D max-pooling as follows:

$$P_{ij}^{l} = \max_{(m,n)\in \text{pooling region}} Z_{(i+m-1)(j+n-1)}^{l} \tag{4.37}$$

Here,

- P_{ij}^{l} is the result of max-pooling at position (i, j) in layer l,
- $Z_{(i+m-1)(j+n-1)}^{l}$ is the activation from position $(i+m-1, j+n-1)$ in layer l,
- The max operation is applied over the defined pooling region.

Finally, the outputs of the pooling layer should be flattened using a flattening layer:

$$\text{Flatten}(P^{l}) = [P_{11}^{l}, P_{12}^{l}, \ldots, P_{mn}^{l}] \tag{4.38}$$

where $\text{Flatten}(P^{l})$ represents the flattened vector of the max-pooled activations.

4.6.2 Hyperparameters tuning

Hyperparameters are external model configuration settings used in ML [21]. Hyperparameters need to be set before the training process starts, in contrast to parameters, which are learned from data during training. Because hyperparameter selection directly affects the model's performance, it is essential. In order to maximize the accuracy, generalizability, and overall performance of the model, hyperparameter tuning entails a methodical investigation of various hyperparameter values. Methods like Bayesian optimization (BO) are frequently used to find the optimal hyperparameter configurations quickly. By improving the model's resilience and flexibility for different datasets, this procedure seeks to increase its predictive power.

A popular method for hyperparameter tuning is BO [22], a probabilistic model-based optimization technique, which will be used in this thesis. The fundamental BO process uses a probabilistic model, usually a Gaussian process (GP), to represent the objective function and direct the hunt for the best hyperparameters (see Figure 4.3). The following is a condensed version of the steps and equations used in BO.

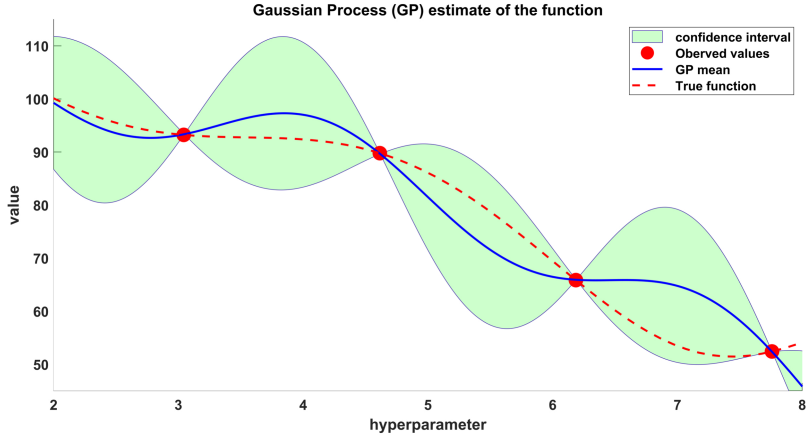


Figure 4.3: Bayesian optimization of a function with the Gaussian process.

Gaussian process model - GP

The objective function $f(x)$ is modeled by the GP as a distribution over functions. The GP predicts the distribution of the objective function for a set of observed data points (x, y),

where x represents the hyperparameter configurations and y represents the corresponding model performance (e.g., validation accuracy or loss):

$$f(x) \sim GP(\mu(x), \sigma^2(x)) \tag{4.39}$$

Acquisition function

The next configuration to be evaluated for hyperparameters is found using the acquisition function $a(x)$. The probability of improvement (PI), expected improvement (EI), and upper confidence bound (UCB) are examples of common acquisition functions:

$$x_{next} = \arg\ \max\ _x a(x) \tag{4.40}$$

Updating the Gaussian process

After evaluating the objective function at the chosen hyperparameter configuration, the GP is updated with the new observation:

$$Update\ GP\ with\ (x_{next}, y_{next}) \tag{4.41}$$

Sequential model-based optimization - SMBO

The overall BO procedure, called sequential model-based optimization (SMBO), assesses the objective function, updates the GP, and iteratively chooses hyperparameter configurations based on the acquisition function:

$$\begin{aligned} x_{\text{next}} &= \arg\max_x a(x) \\ Evaluate\ f(x_{\text{next}}) & \\ Update\ GP\ with\ \left(x_{\text{next}}, f(x_{\text{next}})\right) & \end{aligned} \tag{4.42}$$

The fundamental steps in BO for hyperparameter tuning are represented by these equations. The particulars and formulas can change depending on which probabilistic model and acquisition function are selected during the optimization procedure.

4.6.3 Ensemble learning - EL

After constructing HIs based on individual DL models, an ensemble learning (EL) technique can be employed to create a meta-model capable of addressing inherent randomness in ML models and uncertainties in model structure. This is particularly crucial when the sample size is limited, as in this study. Combining various single prediction models into an EL model (ELM) can effectively leverage the strengths of different base models, thereby enhancing reliability and accuracy [23, 24]. There are three main categories of EL techniques: bagging [25], boosting [26], and stacking [27]. Bagging (e.g., random forest (RF) [28]) involves bootstrapping (random sampling) and aggregation (averaging the base learners' outputs). Boosting (e.g., adaptive boosting (AdaBoost) [29] and extreme gradient boosting (XGB) [30]) combines sequentially arranged base learners. In contrast to the former methods, which use homogenous base models, stacking utilizes heterogeneous base learners and integrates them through training a meta-model (blender).

From another perspective, ELMs based on averaging can be broadly categorized into two types: simple averaging ensemble (SAE) [31] and weighted averaging ensemble

(WAE) [32], with the former being a specific case of the latter. In SAE, the ELM combines predictions from multiple base models. However, a limitation is that each model contributes equally to the ensemble prediction, regardless of its performance. In contrast, WAE assigns weights to ensemble members based on trust in their predictive capabilities, allowing for a more sophisticated contribution.

Despite the fact that a variety of EL methods and models (including RF, least squares linear, support vector machine (SVM), boosting (LSBoost), Bayesian linear regression, GP regression, and Gaussian kernel regression) were investigated, the presented work concentrates on averaging ensemble models using various weighting techniques and DL ensemble models. The decision to use these models over others was made based on their superior performance and effectiveness.

Initially, leave-one-out cross-validation (LOOCV) [33] is employed, where a single unit (composite specimen) is designated for testing, leaving the rest units for subsequent processing. With this in mind, three main dataset divisions can be made:

- (A) considering one test unit without validation (training with a fixed number of epochs);
- (B) considering the test unit itself as validation (intended to prevent overfitting); and
- (C) considering another unit other than the test unit as validation.

Given the small number of specimens available in this study, case B is less generalized yet nonetheless partially valid. In the case of (c), the validation unit is randomly chosen from the left units after taking the test unit out. All three cases are investigated in this work. The base learner models are trained iteratively, typically for 100 repetitions in this study. For case C, the validation set is randomly selected 10 times, and each time, the learning process is conducted with 10 different random seed numbers [34] to initialize weights and biases. Subsequently, the predictions of the 100 HIs generated by the base learner models are ensembled using a process that involves SAE, WAE, and, finally, DL models.

The general WAE can be expressed as follows:

$$f_{WAE} = \sum_{k=1}^{K} \bar{\omega}_k f_k \tag{4.43}$$

where f_k represents the k^{th} individual base learner model and $\bar{\omega}_k$ is its normalized influential weight:

$$\bar{\omega}_k = \frac{\omega_k}{\sum_{k=1}^{K} \omega_k} \tag{4.44}$$

Here, ω_k represents the weight of the k^{th} individual model, which can be determined based on various error metrics. These metrics encompass the mean square deviation (MSE) between the predictions of the k^{th} model ($HI^{k(E)}$, where (E) denotes the ensemble HI) and the simulated HIs (T^k), the model's Fitness (prognostic criteria - Mo, Pr, and Tr), or a combination of both metrics:

$$\omega_k^{MSE} = \frac{1}{MSE\left(T^k, HI^{k(E)}\right)} = \frac{1}{\frac{1}{|v \cup D|}\sum_{j \in (v \cup D)}\left[\frac{1}{N_j}\sum_{i=1}^{N_j}\left(T_i^k - HI_i^{k(E)}\right)^2\right]}; \; k \in (v \cup D) \tag{4.45}$$

$$\omega_k^{Fitness} = a \cdot Mo_{HI} + b \cdot Pr_{HI} + c \cdot Tr_{HI};\ k \in (v \cup D) \tag{4.46}$$

$$\omega_k^{Fitness\_MSE} = \frac{\omega_k^{Fitness} - \min_j\left(\omega_j^{Fitness}\right)}{\max_j\left(\omega_j^{Fitness}\right) - \min_j\left(\omega_j^{Fitness}\right)} + \frac{\omega_k^{MSE} - \min_j\left(\omega_j^{MSE}\right)}{\max_j\left(\omega_j^{MSE}\right) - \min_j\left(\omega_j^{MSE}\right)};\ k, j \in (v \cup D) \tag{4.47}$$

where $|v \cup D|$ is the size of the set. It is important to mention that the units are divided into three portions: training (D), validation (v), and test (τ) sets. Additionally, the training data of D itself is further split into training (D') and validation (v') subsets specifically for the base learner model. In the combined form Eq.4.47, each weight is first scaled in a range $[0, 1]$ to avoid compromising the effectiveness of its scale. If all $\bar{\omega}_k$ values are uniformly set to one, the SAE approach is adopted. After computing weights using Eqs. 4.45-4.47, one can find a subset of superior models by using the weight rankings to increase efficiency and guarantee improved result robustness. The remaining weights are then reset to zero. In this work, the top 10 base learners are kept using Eqs. 4.45 and 4.46, and afterwards, $\omega_k^{Fitness\_MSE}$ and $\bar{\omega}_k$ are calculated.

In addition to averaging ensemble models, another approach involves using a subsequent ML-based model to fuse the predictions, with the goal of reducing the inherent randomness in the base learner models. This ML-based EL model can be implemented using ANN or DL networks. For future reference in the subsequent chapters, relevant architectures and information will be provided if these models are employed in the developed frameworks.

4.7 Conclusions

This chapter has equipped readers with essential knowledge for developing frameworks across diverse applications. Preprocessing techniques, signal processing methods, statistical feature extraction, dimension reduction through PCA-based algorithms, and an overview of deep learning architectures were covered. The introduction of adaptive standardization addressed challenges with conventional techniques. Signal processing methods like FFT, HT, and CEEMDAN were explored for information extraction. Statistical features enriched datasets, and PCA-based algorithms addressed high-dimensional spaces. The deep learning section introduced key architectures, and hyperparameter tuning was discussed. Ensemble learning techniques were presented to enhance prediction robustness. This comprehensive overview lays the groundwork for subsequent chapters.

References

[1] J. Han, J. Pei, and H. Tong, *Data mining: concepts and techniques*. Morgan kaufmann, 2022.

[2] S. Jain, S. Shukla, and R. Wadhvani, "Dynamic selection of normalization techniques using data complexity measures," *Expert Systems with Applications*, vol. 106, pp. 252–262, 2018.

[3] M. Moradi, A. Broer, J. Chiachío, R. Benedictus, T. H. Loutas, and D. Zarouchas, "Intelligent health indicator construction for prognostics of composite structures utilizing

a semi-supervised deep neural network and shm data," *Engineering Applications of Artificial Intelligence*, vol. 117, p. 105502, 2023.

[4] M. Frigo and S. G. Johnson, "Fftw: An adaptive software architecture for the fft," in *Proceedings of the 1998 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP'98 (Cat. No. 98CH36181)*, vol. 3, pp. 1381–1384, IEEE, 1998.

[5] M. Le Van Quyen, J. Foucher, J.-P. Lachaux, E. Rodriguez, A. Lutz, J. Martinerie, and F. J. Varela, "Comparison of hilbert transform and wavelet methods for the analysis of neuronal synchrony," *Journal of neuroscience methods*, vol. 111, no. 2, pp. 83–98, 2001.

[6] G. Zhao, B. Wang, T. Wang, W. Hao, and Y. Luo, "Detection and monitoring of delamination in composite laminates using ultrasonic guided wave," *composite Structures*, vol. 225, p. 111161, 2019.

[7] N. E. Huang, Z. Shen, S. R. Long, M. C. Wu, H. H. Shih, Q. Zheng, N.-C. Yen, C. C. Tung, and H. H. Liu, "The empirical mode decomposition and the hilbert spectrum for nonlinear and non-stationary time series analysis," *Proceedings of the Royal Society of London. Series A: mathematical, physical and engineering sciences*, vol. 454, no. 1971, pp. 903–995, 1998.

[8] Z. Wu and N. E. Huang, "Ensemble empirical mode decomposition: a noise-assisted data analysis method," *Advances in adaptive data analysis*, vol. 1, no. 01, pp. 1–41, 2009.

[9] M. E. Torres, M. A. Colominas, G. Schlotthauer, and P. Flandrin, "A complete ensemble empirical mode decomposition with adaptive noise," in *2011 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, pp. 4144–4147, IEEE, 2011.

[10] G. H. Dunteman, *Principal components analysis*, vol. 69. Sage, 1989.

[11] E. De Moura, C. d. R. Souto, A. A. Silva, and M. A. d. S. Irmão, "Evaluation of principal component analysis and neural network performance for bearing fault diagnosis from vibration signal processed by rs and df analyses," *Mechanical Systems and Signal Processing*, vol. 25, no. 5, pp. 1765–1772, 2011.

[12] L. Van Der Maaten, E. O. Postma, H. J. van den Herik, *et al.*, "Dimensionality reduction: A comparative review," *Journal of Machine Learning Research*, vol. 10, no. 66-71, p. 13, 2009.

[13] Y. He, M. Li, Z. Meng, S. Chen, S. Huang, Y. Hu, and X. Zou, "An overview of acoustic emission inspection and monitoring technology in the key components of renewable energy systems," *Mechanical Systems and Signal Processing*, vol. 148, p. 107146, 2021.

[14] B. De Ketelaere, M. Hubert, and E. Schmitt, "Overview of pca-based statistical process-monitoring methods for time-dependent, high-dimensional data," *Journal of Quality Technology*, vol. 47, no. 4, pp. 318–335, 2015.

[15] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting," *The journal of machine learning research*, vol. 15, no. 1, pp. 1929–1958, 2014.

[16] R. H. Hahnloser, R. Sarpeshkar, M. A. Mahowald, R. J. Douglas, and H. S. Seung, "Digital selection and analogue amplification coexist in a cortex-inspired silicon circuit," *nature*, vol. 405, no. 6789, pp. 947–951, 2000.

[17] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[18] R. Yang, S. K. Singh, M. Tavakkoli, N. Amiri, Y. Yang, M. A. Karami, and R. Rai, "Cnn-lstm deep learning architecture for computer vision-based modal frequency detection," *Mechanical Systems and signal processing*, vol. 144, p. 106885, 2020.

[19] A. Graves, S. Fernández, and J. Schmidhuber, "Bidirectional lstm networks for improved phoneme classification and recognition," in *International conference on artificial neural networks*, pp. 799–804, Springer, 2005.

[20] Y. LeCun, Y. Bengio, *et al.*, "Convolutional networks for images, speech, and time series,"

[21] M. Feurer and F. Hutter, "Hyperparameter optimization," *Automated machine learning: Methods, systems, challenges*, pp. 3–33, 2019.

[22] J. Snoek, H. Larochelle, and R. P. Adams, "Practical bayesian optimization of machine learning algorithms," *Advances in neural information processing systems*, vol. 25, 2012.

[23] D. Opitz and R. Maclin, "Popular ensemble methods: An empirical study," *Journal of artificial intelligence research*, vol. 11, pp. 169–198, 1999.

[24] R. Polikar, "Ensemble based systems in decision making," *IEEE Circuits and systems magazine*, vol. 6, no. 3, pp. 21–45, 2006.

[25] L. Breiman, "Bagging predictors," *Machine learning*, vol. 24, pp. 123–140, 1996.

[26] J. Friedman, T. Hastie, and R. Tibshirani, "Additive logistic regression: a statistical view of boosting (with discussion and a rejoinder by the authors)," *The annals of statistics*, vol. 28, no. 2, pp. 337–407, 2000.

[27] D. H. Wolpert, "Stacked generalization," *Neural networks*, vol. 5, no. 2, pp. 241–259, 1992.

[28] L. Breiman, "Random forests," *Machine learning*, vol. 45, pp. 5–32, 2001.

[29] Y. Freund and R. E. Schapire, "A desicion-theoretic generalization of on-line learning and an application to boosting," in *European conference on computational learning theory*, pp. 23–37, Springer, 1995.

[30] T. Chen and C. Guestrin, "Xgboost: A scalable tree boosting system," in *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining*, pp. 785–794, 2016.

[31] S. Haykin, *Neural networks: a comprehensive foundation*. Prentice Hall PTR, 1998.

[32] S. Hashem, "Optimal linear combinations of neural networks," *Neural networks*, vol. 10, no. 4, pp. 599–614, 1997.

[33] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*, vol. 2. Springer, 2009.

[34] M. Matsumoto and T. Nishimura, "Mersenne twister: a 623-dimensionally equidistributed uniform pseudo-random number generator," *ACM Transactions on Modeling and Computer Simulation (TOMACS)*, vol. 8, no. 1, pp. 3–30, 1998.

5

Designing HI for T-Stiffener Composite Panels using Acoustic Emission

This chapter presents three frameworks integrating AI and signal processing to formulate health indicators (HIs) for T-Stiffener composite panels using the acoustic emission technique. The frameworks leverage the SSL paradigm, two-stage machine learning for spatial and temporal information extraction, physics-based Bayesian optimization, and SS ensemble deep learning.

5.1 Introduction

In this chapter, three frameworks are introduce for designing HIs tailored to T-Stiffener composite panels through the utilization of acoustic emissions (AE). The frameworks are as follows:

1. **FFT-SSLSTM:** Semi-Supervised Long Short-Term Memory (fed by time's and frequency's features)
2. **FFT-PCA-2S-ML-PBO:** Two-Stage Machine Learning Approach with Physics-based Bayesian Optimization (following PCA-based dimension reduction upon time's and frequency's features)
3. **CEEMDAN-SSEDL:** Semi-Supervised Ensemble Deep Learning (fed by intrinsic mode functions of CEEMDAN)

AE data were collected from twelve single-stiffener composite panels under compression-compression (C-C) fatigue loading [1–3]. This chapter details the experimental campaigns, followed by a dedicated exploration of the specialized data acquisition and pre-processing techniques tailored for the AE technique in Section "AE Data Acquisition and Pre-processing." Subsequently, the three frameworks are expounded upon in detail, with their results comprehensively presented in their respective sections. A comparative analysis of the main results from each framework is conducted and discussed in Section "Discussion." To conclude, a summary of the findings is presented in the concluding remarks within the "Conclusions" section.

5.2 Experimental Campaigns

As part of the H2020 ReMAP project, two test campaigns were held at the Delft Aerospace Structures and Materials Laboratory (DASML) in 2019 and 2020, in which twelve composite skin-stiffener panels were tested under C-C fatigue loading. The panels are made up of a skin panel and a single T-stiffener according to an Embraer design (see Figure 5.1). The skin and stiffener are all made of IM7/8552 carbon fiber-reinforced epoxy unidirectional prepreg with layups of $[45/-45/0/45/90/-45/0]_S$ and $[45/-45/0/45/-45]_S$, respectively [3]. Two resin blocks for each single-stiffened composite panel (SSP) were also included to ensure that the load was distributed evenly. The dimensions of one panel are shown in Figure 5.1. An initial damage in the form of an artificial disbond (Teflon insert during manufacturing) in the skin//stiffener interface with different sizes or an impact damage of around 10 J located on the stiffener area is introduced to some panels. This was done in order to create a damaged area and monitor its growth over the course of the experiments. For the panels that do not experience impact prior to testing, after 5000 cycles, the impact is performed, even for the panels with artificial disbond defects. These factors simulate various realistic and uncertain phenomena in the experiments, resulting in a wide range of EoL from 48.7K to 756.3K cycles, which will make it more challenging to perform HI construction and RUL prediction (more information in Table 5.1).

The damage growth in the panel was monitored using six different techniques: (1) AE, (2) distributed fiber optic sensing (DFOS), (3) fiber Bragg gratings (FBG) (only for the campaign 2019), (4) lamb wave (LW) (the last two were only available in the first

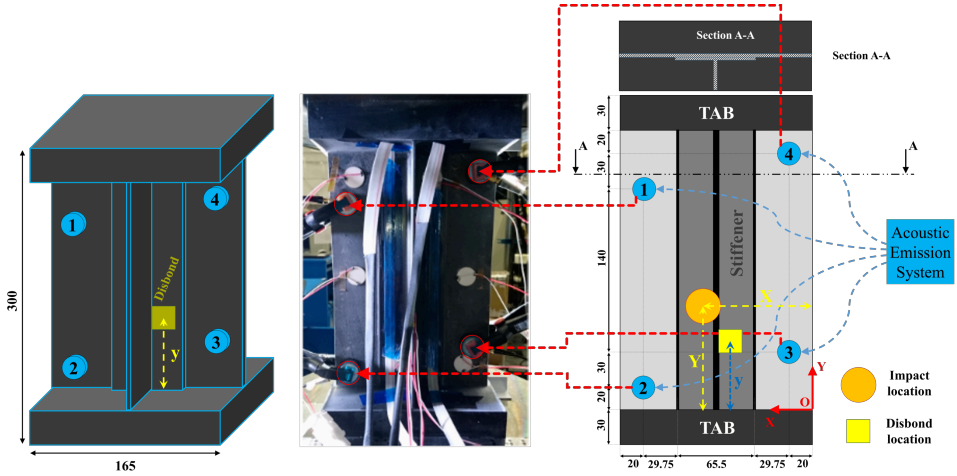


Figure 5.1: Single stiffener panel: (a) 3D view, (b) stiffener side, (c) sensor coordinates, where the four AE sensors shown as blue circles, impact and disbond locations shown as a movable orange circle and yellow rectangle for different panels (all dimensions in [mm]).

5

Table 5.1: The information of the composite specimens tested under C-C fatigue loading.

Year	Campaign 2019			Campaign 2020								
Name	L1-03	L1-05	L1-09	L1-49	L1-50	L1-51	L1-52	L1-54	L1-55	L1-56	L1-59	L1-60
Composite Specimen	1	2	3	4	5	6	7	8	9	10	11	12
X-location of impact (mm)	50	115	82.5	50	50	50	50	50	115	50	50	115
Y-location of impact (mm)	80	160	140	160	160	160	160	160	80	160	160	80
Time of Impact	at 0 cycles	at 0 cycles	at 0 cycles	after 5000 cycles	after 5000 cycles	after 5000 cycles	at 0 cycles	at 0 cycles	after 5000 cycles	after 5000 cycles	after 5000 cycles	after 5000 cycles
Size of disbond (mm)				15×20	20×20	20×25						
y-location of disbond (mm)				60	60	60						
Min Load (kN)	-6.5	-6.5	-6.5	-6.5	-6.5	-6.5	-6.5	-6.5	-6.5	-6.5	-6.5	-6.5
Max Load (kN)	-65	-65	-65	-65	-65	-65	-65	-65	-65	-65	-65	-65
Cycles (MTS)	152,458	144,969	133,281	48,702	65,500	94,431	368,558	510,961	226,356	756,226	110,137	170,884
Labeled Cycles	152,457	144,970	133,282	48,703	65,502	94,437	368,590	510,982	226,361	756,264	110,185	170,898
Error in cycles labeling	-1	1	1	1	2	6	32	21	5	38	48	14

campaign), (5) digital image correlation (DIC), and (6) camera. Only data from the AE technique is analyzed for the purposes of this chapter.

The panels were loaded in C-C fatigue loading using an MTS machine with a frequency of 2 Hz and an R-ratio of 10 which means that the fatigue load was set in a compression load range of [6.5, 65.0] kN. Although the R-ratio was intended to hold invariant, the panels experienced a loss in load-bearing capacity. The fatigue load was disrupted at regular intervals of 500 cycles to allow the SHM systems to take measurements. All data are publicly available in [1, 2]. Table 5.1 summarizes the aforementioned explanations and provides additional details.

5.3 AE Data Acquisition and Pre-processing

The AE sensors used are Vallen Systeme GmbH VS900-M broadband sensors with a frequency range of 100–900 kHz. The AE hits were recorded using an AMSY-6 Vallen

acquisition system. Moreover, Vallen preamplifiers with a gain of 34 dB were used. Four AE sensors were clamped in various positions on the skin of the panels to create a parallelogram, enabling to localize damage and to obtain a quantification of the location uncertainty. The AE Sensors 1, 2, 3, and 4 had [x, y] locations of [145.0, 190.0], [145.0, 20.0], [20.0, 50.0], and [20.0, 220.0] mm, respectively, as seen in Figure 5.1(right). As multiple sensing techniques were employed for damage monitoring in the SSP, the AE sensor positions were selected through a trade-off with those sensor positions of the other techniques, in particular to maximize the monitoring region of both the AE and LWDS techniques. An amplitude threshold of 60 dB was set for capturing the hits to avoid the recording of noise signals. Only events localized within the AE sensor area are taken into account. The internal Vallen processor for planar positioning, which is based on Geiger's model [4], was used for localization. A filter was also used to exclude events with a position uncertainty greater than 50 mm. More detailed information for the applied localization method to AE data are described in subsection of Localizing data. Six variables (low-level features) containing amplitude (A), rise time (R), energy (E), counts (CNTS), duration (D), and RMS have been extracted and recorded from AE events (see Figure 5.2(a)).

In the following, this section describes the pre-processing procedures, which include labeling cycles, localizing data, windowing, and missing values. Another pre-processing step that should be conducted after feature extraction is a standardization (zero-mean) technique.

5.3.1 Labeling cycles

Labeling cycles on AE data is required for this study since a SSL with some hypothetical HIs as targets is proposed in order to construct HI, and the time (cycle) of each acquired AE event must be known in order to generate these targets. Due to the constraints of the MTS machine's output channels and the AE system's input channels and software, the AE system is unable to directly record cycles from the MTS machine. Nonetheless, since the AE system and MTS machine have been synchronized, and the AE system can import displacement and load values from MTS machine next to the other six variables from AE sensors, signal processing methods can approximate the cycle number of each hit, in which (cycle) that hit plus possibly more other hits occurred. Table 5.1 shows the number of cycles reported by the MTS machine (exact) and of the labeled cycles through the load signal (approximate), as well as the error between them. Given that the maximum error percentage is 0.044% (48/110137 for specimen 11), the estimated labeled cycles provided with the AE variables can be used.

5.3.2 Localizing data

Geiger's method [4] was used to localize the AE data (Figure 5.2(b)), and it allowed for planar localization of the AE events throughout the fatigue testing. This method assumes a constant wave velocity in all directions, which was determined using Hsu-Nielsen sources on pristine specimens [5]. The wave speed was determined in both the x-direction (4423 m/s) and y-direction (6107 m/s), and the mean wave speed was then calculated as 5265 m/s. This was used as an input to Geiger's method to determine the planar location of the AE events. Since Geiger's method is a time-of-arrival approach, its application in anisotropic composite specimens can lead to errors in the AE event localization. The application of

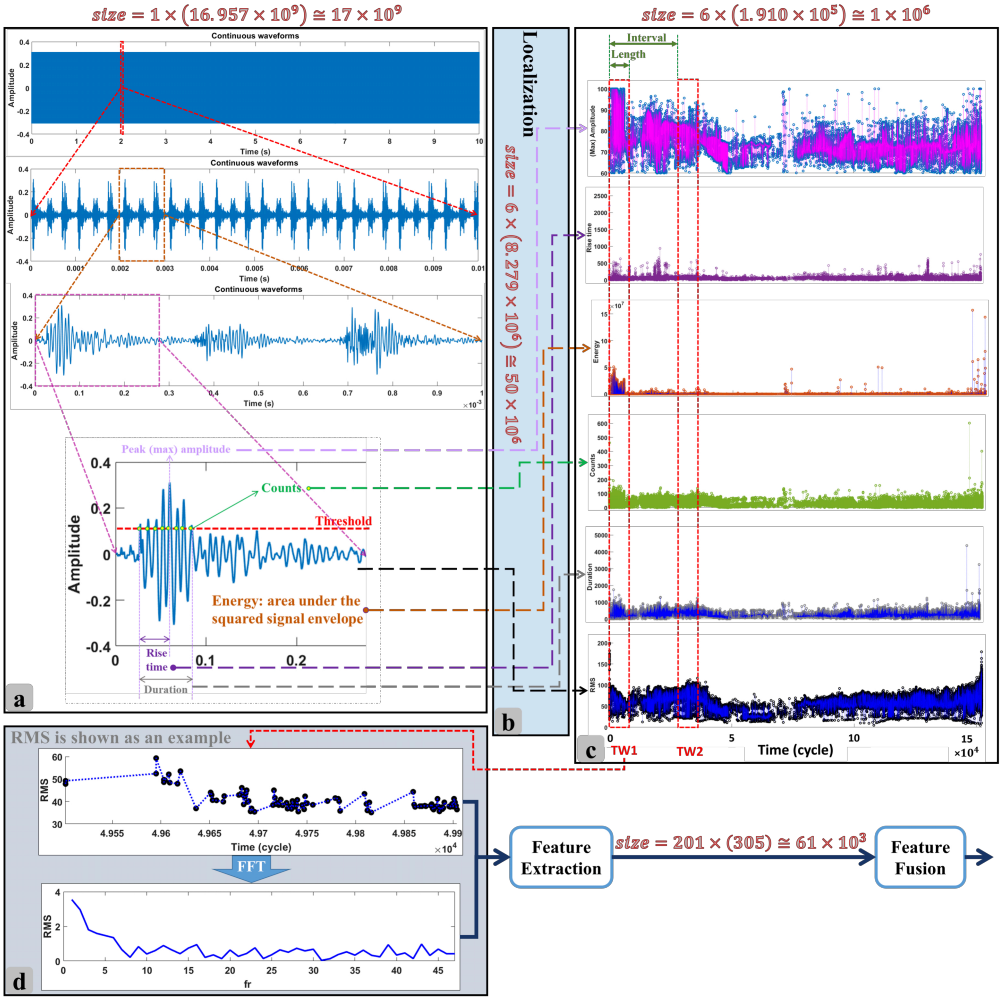


Figure 5.2: Data reduction during (a) pre-processing (initial feature extraction), (b) localization, (c) windowing, and (d) signal processing.

four AE sensors allows for the calculation of this position error, and a filter was implemented to exclude events with a position uncertainty larger than 50 mm. Lastly, events outside the AE sensor region are filtered.

5.3.3 Windowing

In the third step of pre-processing, as can be seen in Figure 5.2(c), the signals (AE variables extracted from waveforms, including amplitude, rise time, duration, energy, counts, and RMS) are windowed for two reasons: one is that memorizing and analyzing all data from the beginning to the current time costs a tremendous computation time; another is that analyzing and comparing data at a single instant without taking into account nearby

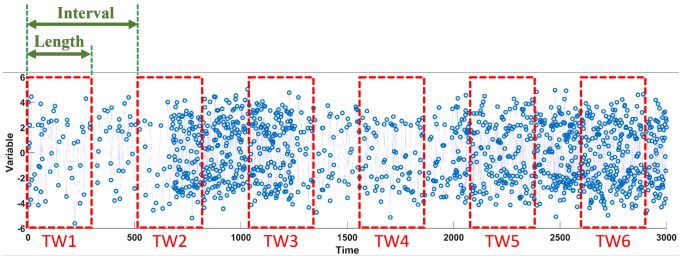


Figure 5.3: Six time windows resulted by windowing process on artificial data with a constant length and interval.

time steps is insufficient, especially for nonstationary signals. It should be noted that some SHM techniques, like AE, do not record data points at a constant rate since they are passive techniques and depend on the number of events occurring in the structure. For example, AE might measure 50 events in the first 10 seconds while it might measure 1000 events in the next 10 seconds. As can be seen in Figure 5.3, the two main factors in the windowing process are the length of each window and the interval between two sequential windows, which are essential and important since they influence the final results and decisions. These factors might also be considered in a dynamic way rather than the static one and can be optimized as such. The windowing process for the current study was cycle-based, with a static length and interval of 500 cycles due to the natural interval of QS loads.

5.3.4 Missing values

Since no events might have been recorded in a few intervals of 500 cycles due to the applied filters, there are missing values for those time windows. Because missing values have an impact on subsequent phases of the HI construction process, they should be eliminated or filled in, with the first option being taken for the windowed signals. Also, after the feature extraction step, some statistical features may be missed. For this step, linear interpolation is used to fill in the missing values.

5.4 1^{st} framework: FFT-SSLSTM

The 1^{st} framework developed upon AE data is shown in Figure 5.4, where the first block is already explained in the previous section. The second and third blocks—SP by FFT and FE by statistical features—are also explained in the former chapter.

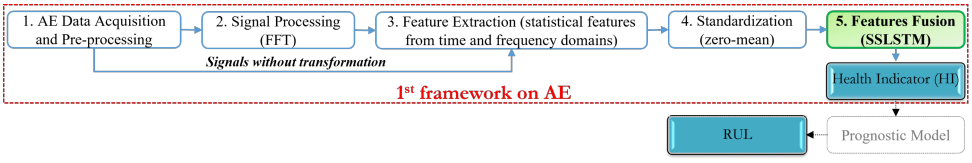


Figure 5.4: Workflow of the 1^{st} framework developed upon AE data, including SSLSTM as the feature fusion step.

Feature extraction: The statistical features in the time and frequency domains have

been extracted based on Tables 4.1 and 4.2. As a result, 33 features are extracted from each of the 6 windowed signals (low-level features) of the AE data, including amplitude, rise-time, energy, counts, duration, and RMS. The broad features field has been expanded to include three additional possible useful features: cumulative Rise-time/Amplitude ratio, cumulative energy, and cumulative counts [6]. The AE dataset yielded a total of 201 features (6×33 + 3). It should be noted that FE procedure may also be regarded as a dimension reduction step (Figure 5.2), as raw signals with thousands of data points within each window have been reduced to 201 data points. In fact, data with billions of records has been reduced to thousands.

Standardization (zero-mean): The input features are standardized before being input into the network using a zero-mean normalization technique (Chapter 4) that used only the training dataset's mean value and standard deviation.

5.4.1 Feature fusion: semi-supervised LSTM (SSLSTM)

Built upon the SSL paradigm, which serves as the central concept elucidated in Chapter 3, an ANN architecture named SSLSTM has been developed. SSLSTM is designed to fuse features and is composed of four key types of layers: FC, dropout, ReLU, LSTM, and regression layers (refer to Figure 5.5). Notably, the training data underwent a sequence length-based sorting process to minimize the need for excessive padding in the batches.

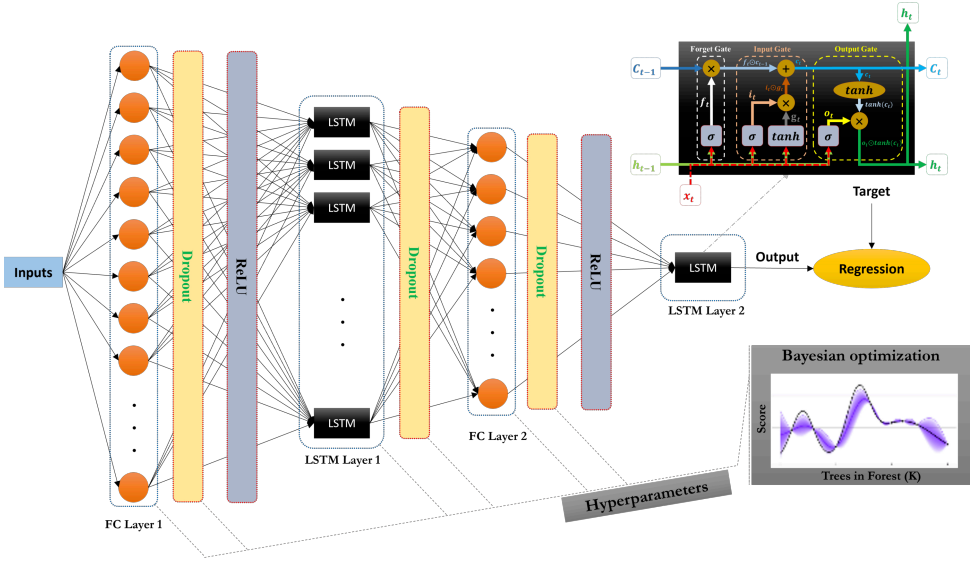


Figure 5.5: Multi-layers LSTM network proposed for feature fusion.

5.4.2 Hyperparameters optimization and model validation

After fixing an acceptable configuration of the neural network layers (Figure 5.5), a BO algorithm was used to set the hyperparameters, including the number of neurons at each layer, batch size, and dropout.

For this framework, two main and trustworthy validation methods in the ML fields,

which are LOOCV and holdout validation methods [7], are employed. First, using the holdout method with 10, 1, and 1 units as training, validation, and test datasets, respectively, the BO algorithm was used to find the top hyperparameters' sets and models (with the maximum RMSE over all units as the objective function which must be minimized). It should be noted that by considering the maximum RMSE of all units rather than other statistical parameters like the mean value of RMSE of them, the optimization algorithm attempts to simultaneously decrease the mean value and standard deviation of RMSE, which is more desirable. Then, the LOOCV method, having 11 and 1 units as training and validation dataset, respectively, was applied to the top 10 models obtained by the holdout validation and BO to check models' performance for the other folds, with 10 replications. Finally, the performance of these models will be described as a distribution with a mean expected error and a standard deviation.

5.4.3 Results

The deep learning framework and signal processing parts were developed using MATLAB R2021a; a high performance computing cluster (Beowulf style) with 12 processors on one node for the BO algorithm, and a laptop with an Intel Core i7-8665U CPU and 16 GB RAM for training the DL networks and the other parts (such as pre-processing and signal processing) were used. In this section, following the results of the holdout validation and the LOOCV, the best-proposed model will be discussed in comparison to the relevant literature in the subsection Discussion.

Holdout validation

First, the holdout method has been used to validate the model, with the first ten SSPs for training, the 11^{th} SSP for validation, and the 12^{th} SSP for the test dataset.

An Adam optimizer [8] was used to learn the DL model, with an initial learning rate of 0.005, a learning rate drop factor of 0.2, a learning rate drop period of 5, and a gradient threshold of 1, which all have been selected after trial and error. Before each epoch, the training dataset was shuffled. Despite the fact that the maximum number of training epochs was set to 500, the network's output is based on the best validation loss, with the validation check frequency set to 30 iterations (number of trained batches) and validation check patience set to 6.

The BO algorithm was given 120 trials in parallel computing to optimize the hyperparameters. The number of neurons in the FC layer 1 and FC layer 2 as well as the number of units in the LSTM layer 1 have been allocated [1,201], [1,50], and [1,256], respectively, based on trial and error. It is worth noting that the LSTM layer 2 only contains one unit. For dropout, the interval [0,0.5] quantized to 0.1 was also examined. Since the training dataset comprises ten units, the interval [1,5] quantized to one has been explored for batch size. Since each set of BO final results is also dependent on the initial start points, the entire procedure was repeated several times. The top 10 hyperparameter sets (models) are presented in Table 5.2. As can be seen, the varied configurations have resulted in quite close RMSE ranged [0.08-0.11], which is the maximum RMSE over all units as the objective function of the BO algorithm.

Figure 5.6(a) shows (merely as a case chosen to display the intuitive results) the constructed HIs by model 1, which is the first ranked, and their RMSE can be seen in Fig-

Table 5.2: The top 10 hyperparameter sets (models) obtained by the BO algorithm and holdout validation with the 11^{th} unit as the validation and the 12^{th} unit as the test dataset. RMSE is the maximum one over all units.

Model (rank)	Batch Size	Dropout	FCL1	LSTM1	FCL2	RMSE
1	4	0.3	110	154	50	0.0829
2	5	0.4	124	83	48	0.0884
3	5	0.5	201	79	36	0.0983
4	5	0.4	152	81	27	0.1013
5	5	0.5	41	142	43	0.1026
6	2	0	30	256	45	0.1052
7	5	0.4	124	56	48	0.1055
8	5	0.1	137	20	39	0.1067
9	5	0.4	161	92	48	0.1084
10	5	0.4	53	120	50	0.1110

ure 5.6(b). The RMSE for SSPs 1, 11, and 12 slightly diverged from the mean value of RMSE for all units, which is 4%. Since comparing the quality of the constructed HIs based solely on RMSE could not provide completely applicable information from a prognostic standpoint, the prognostic criteria Mo, Tr, and Pr, as well as their sum given all units (Fitness by Eq. 3.4), are shown in Figure 5.7 for all the individual input extracted features as well as the HIs constructed by the model 1. The top four features with a Fitness score higher than 1.5 are features 185, 184, 88, and 183, respectively.As can be seen, the high Fitness score of 2.891 for HIs, which is 77.3% higher than the best feature (1.630), demonstrates the high efficiency of the model 1 to construct HIs following the prognostic criteria. In fact, this Fitness improvement represents the performance of the proposed scenario and the whole developed algorithm because the model, the proposed DNN architecture, might still be enhanced by adding and/or changing characteristics such as the other types of layers, units, neurons, activation functions, and hyperparameters. However, as previously discussed, the main focus in the current research is how to implement the prognostic criteria in the process of HI construction. When the overall implementation methodology of the prognostic criteria has been validated, other enhancements like various optimization methods or DNN architectures can be studied in the main proposed roadmap. Nevertheless, the models are investigated in accordance with LOOCV in the next subsection, due to the shortcomings of the holdout validation in evaluating the generalization of the DNN models.

LOOCV

The ten models listed in Table 5.2 are tested with 10 repetitions on the 12 folds of LOOCV. It should be noted that the fold i refers to the fact that the i^{th} unit is the validation and the rest are training datasets. Figure 5.8 shows the mean value and standard deviation of RMSE calculated over these repetitions for only the test dataset (e.g. for fold 1, unit 1 is the test dataset). In other words, the training (SSPs) datasets were not taken into account during calculation of the mean value and standard deviation. For example, in the first fold of LOOCV, the 1st unit was considered the test, and units 2-12 were considered for training, and when one model had completed training, the network was tested on the 1^{st} unit. The mean value and standard deviation for that model and that fold (only for the test dataset which in this example is the 1^{st} SSP) were then calculated over ten repeats. This

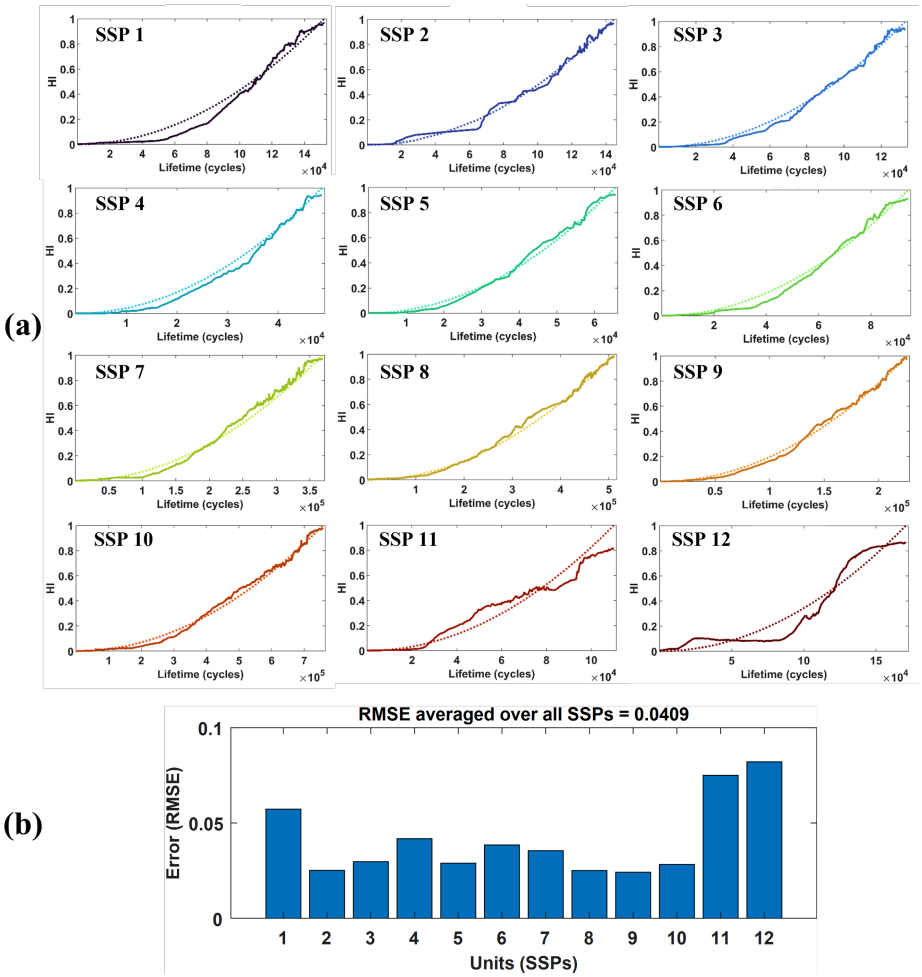


Figure 5.6: (a) HIs constructed by model 1 and (b) their RMSE. The SSP 11 and 12 are the validation and test datasets, and the rest are the training dataset. Dot lines are the target HIs.

process has been performed for each of the 12 folds and each of the 10 best models from Table 5.2.

Figure 5.8 demonstrates that some SSPs, such as 3, 5, 7, 8, and 12, have better performance for all models; however, some, such as 1 and 6, suffer from randomness in the DNN algorithm, which is owing to the stochastic nature of ANN and randomness in the experimental data. Figure 5.9 depicts a line plot of the mean value of RMSE with error (standard deviation) bars for all folds, illustrating a measurement of the generalization of the models. According to this figure, models 8, 2, and 7 are the best generalized ones with mean RMSE value of 0.121±0.090, 0.133±0.082, and 0.139±0.085, respectively.

As previously stated, the Fitness scores could be more appropriate to report due to the deficiency in RMSE from a prognosis aspect. This can be performed in two ways:

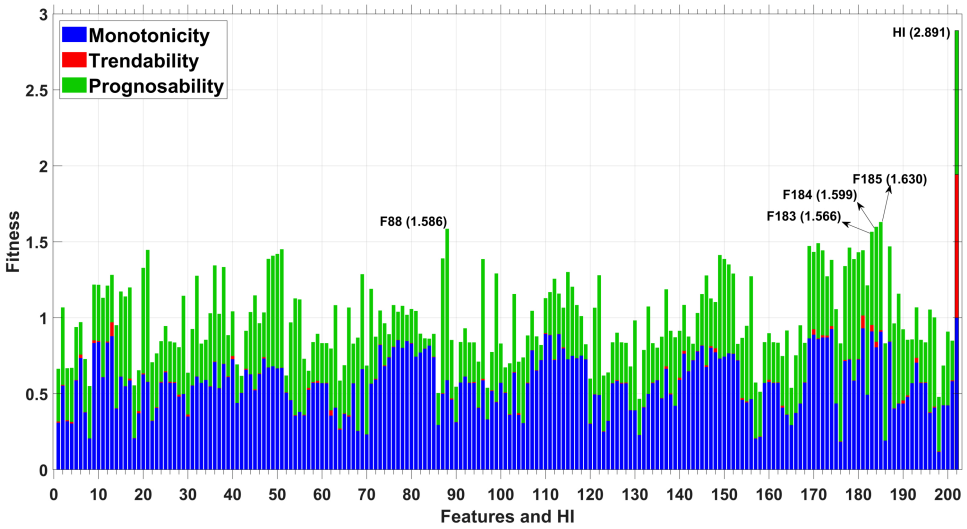


Figure 5.7: The prognostic criteria for all 201 extracted features and the constructed HIs for model 1.

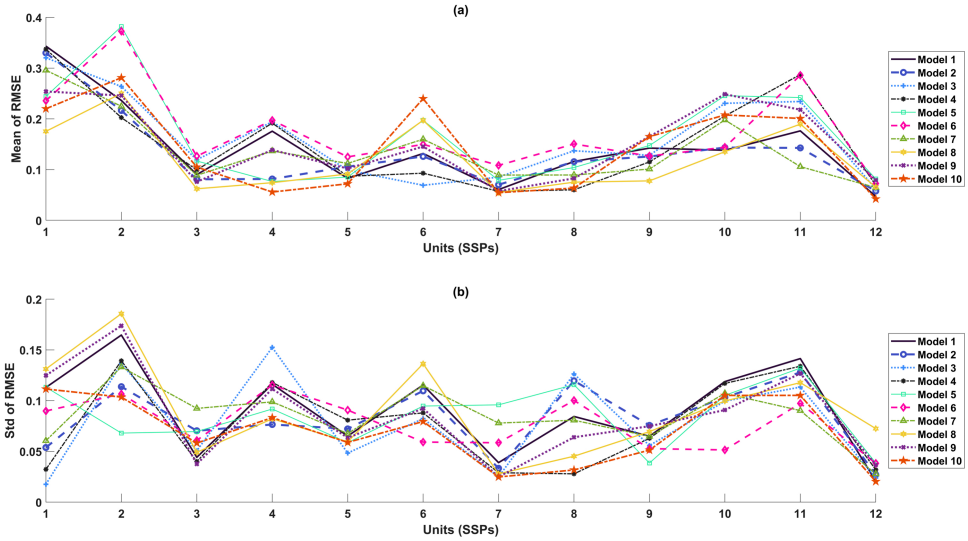


Figure 5.8: (a) Mean value and (b) standard deviation of RMSE of test datasets (SSP 1 to SSP 12) over 10 repetitions for the top ten models.

1. The Fitness score of the constructed HIs is measured for each replication; next, it is averaged across all replications; and finally, the findings for various models and folds are presented (Table 5.3 and Figure 5.10).

2. After all replications have been completed, the constructed HIs for each SSP are averaged across all replications, and the Fitness score of the averaged HIs is calculated. Finally, the outcomes for various models and folds are provided (Table 5.4

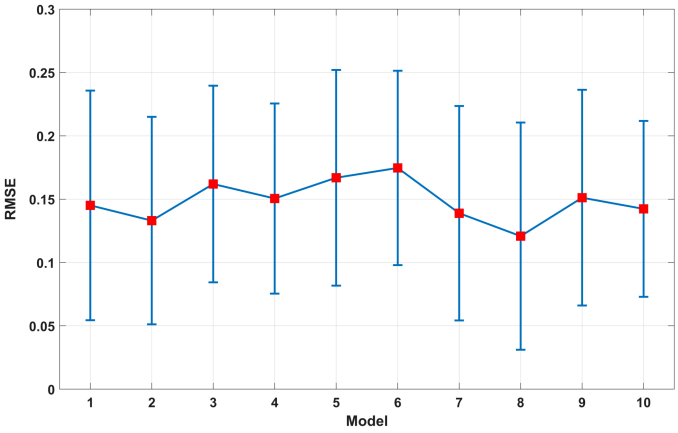


Figure 5.9: Mean value of RMSE over all folds' tests (LOOCV) and after 10 replications, with error (Std) bars for the top ten models.

and Figure 5.12). This could be a case of models making ensemble predictions.

Each approach is discussed in more detail next. It should be noted that the Fitness scores for all extracted features remain unchanged and are identical to those in Figure 5.7.

According to Table 5.3, the most challenging fold for the models to learn is the first one, with a mean Fitness value of 1.550±0.216, and the other two worse folds are 10 and 2, with mean Fitness values of 1.780±0.212 and 1.871±0.139, respectively, and the rest have mean Fitness values of more than 2. The best generalized models are 2, 7, and 1 in order, with mean Fitness values of 2.360±0.415, 2.342±0.425, and 2.337±0.427, respectively. To better compare different models and folds, the distribution of the average Fitness value of HIs for various folds and models can be seen in Figure 5.10.

Table 5.3: Averaged Fitness scores over 10 repetitions of the constructing HI.

Model (rank)	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Fold 11	Fold 12
1	1.3715	1.9550	2.7984	2.0050	2.7291	2.2615	2.7084	2.5093	2.5791	2.0477	2.1968	2.8804
2	1.3663	2.0114	2.7561	2.4697	2.6075	2.1736	2.7587	2.5961	2.5629	1.9182	2.2626	2.8402
3	1.2923	1.8778	2.7210	1.9112	2.6438	2.5747	2.5844	2.6227	2.5624	1.5305	1.8487	2.7557
4	1.3048	1.9465	2.7622	1.8872	2.7150	2.3718	2.7805	2.7565	2.6879	1.7061	1.8616	2.7874
5	1.7903	1.5186	2.7209	2.5297	2.6217	1.7716	2.6271	2.6974	2.4861	1.6657	1.8323	2.7910
6	1.8221	1.7951	2.6852	1.9317	2.5055	1.9714	2.4565	2.4090	2.6288	2.0745	1.7897	2.6872
7	1.3730	1.9365	2.7382	2.3269	2.6166	2.1636	2.5459	2.5989	2.6861	1.7844	2.5063	2.8254
8	1.8142	1.9864	2.7427	2.4059	2.5146	1.7729	2.7520	2.6329	2.6575	2.0011	1.9362	2.6690
9	1.6320	1.9206	2.8524	2.1958	2.7016	2.2784	2.7852	2.7099	2.5107	1.4313	1.8569	2.7445
10	1.7323	1.7640	2.7064	2.4361	2.7218	1.6098	2.7767	2.7301	2.3921	1.6370	2.0289	2.8285
* "Green color → Red color" equalizes "Best result → Worst result"										0	1.5	3

Figure 5.10(a) demonstrates that the models can appropriately construct HIs for folds 3, 5, 7, 8, 9, and 12 with a Fitness value greater than 2.575±0.090 (fold 9). The remaining folds are affected by the model's low mean value or/and high variance. Model 6 has the lowest Fitness value averaged over all folds (2.230±0.349), but it has the lowest variance (see Figure 5.10(b)). The highest average Fitness value pertains to model 2 by which the constructed HIs in iteration 8 (best one) can be seen in Figure 5.11 It is worth noting that the HIs for all units shown in this figure are from test datasets corresponding to relevant

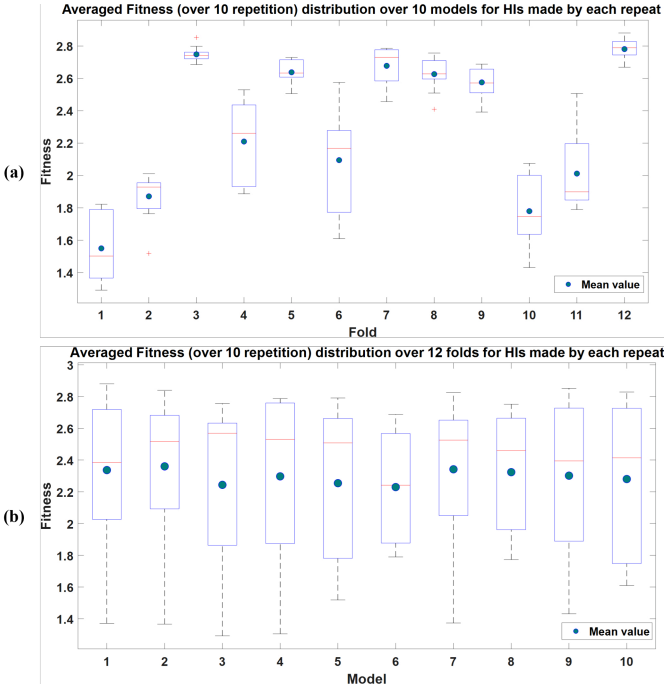


Figure 5.10: The distribution of the averaged (across all replications) Fitness value of HIs for various (a) folds and (b) models.

folds, and there are no constructed HIs from training datasets. Model 2 has not adequately learned SSP 1, and also SSPs 6, 10, 2, and 9 are not as qualified as the rest which have considerably good agreement.

So far, the prognostic criteria were averaged over HIs constructed by ten iterations. Hereinafter, the HIs constructed by ten iterations are averaged, and then, the HIs after simple averaging ensemble (SAE) are investigated to report the prognostic criteria (Fitness by Eq. 3.4).

Table 5.4 shows that the first fold is again the most challenging for the models to learn, with a mean Fitness value of 2.019±0.366, which is substantially better than what was reported in the previous state (Table 5.3), while the remainder have mean Fitness values of higher than 2.5. The best generalized models are 7, 8, and 5 in order, with mean Fitness values of 2.786±0.144, 2.747±0.146, and 2.729±0.199, respectively. The average (ensemble) HIs obviously conform better to the prognostic criteria.

The distribution of the Fitness value of the ensemble HIs by SAE can be seen in Figure 5.12 to better compare various models and folds. Figure 5.12(a) indicates that the models can construct HIs quite effectively for all folds except 1, 10, 6, and 2 when compared to the rest, in which the first fold with the lowest mean value and highest variance of Fitness is severe and distinguishable. The best Fitness value distribution pertains to model 7 by which the constructed ensemble HIs (by SAE over all iterations) can be seen in Figure 5.13(a). The discrepancy in deviation between the target, which is the ideal hypothet-

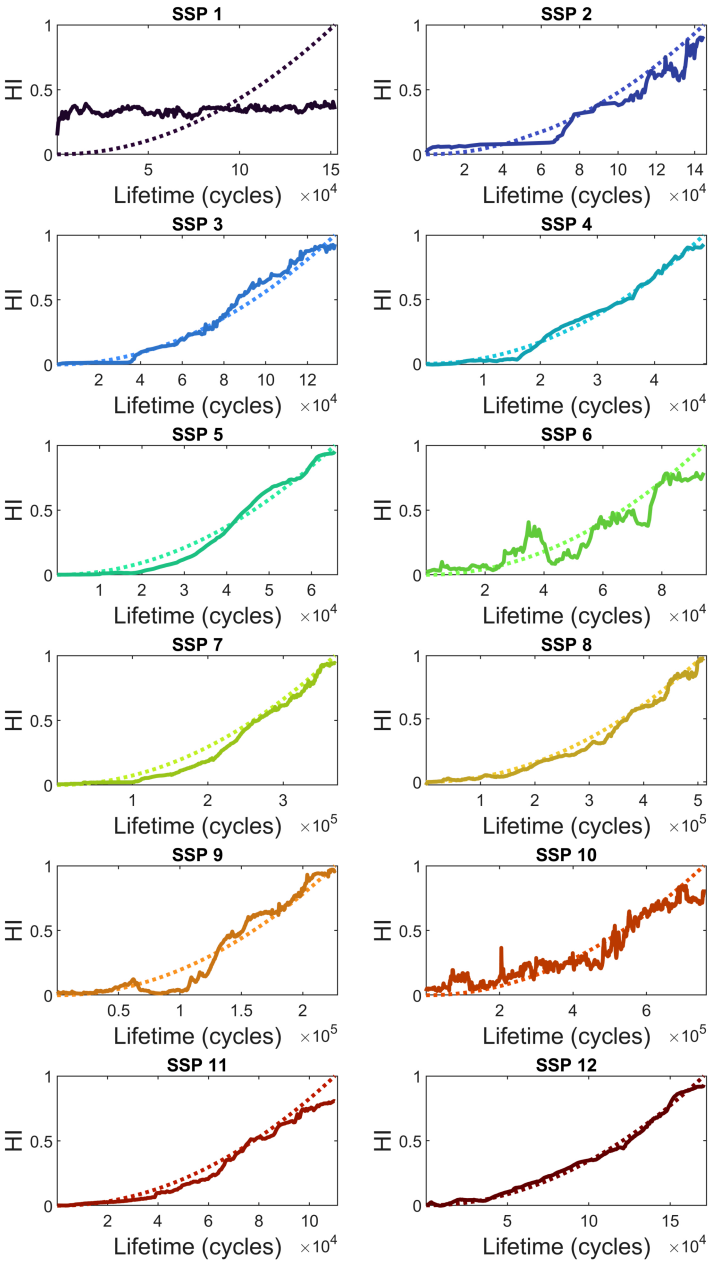


Figure 5.11: HIs constructed by model 2 (in iteration 8 - best one). All shown HIs derive from test datasets matching to relevant folds, not training datasets, for all samples. Dot lines are the ideal HIs.

ical HI, and the average constructed HI for the first fold (SSP 1) is remarkable. Therefore, this fold containing its training dataset (SSPs 2 to 12) has been shown in Figure 5.13(b).

Table 5.4: Fitness scores of the ensemble HIs by SAE over 10 repetitions.

Model (rank)	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Fold 11	Fold 12
1	2.0168	2.6074	2.9163	2.6504	2.8916	2.6407	2.8842	2.8194	2.8263	2.5998	2.8319	2.9233
2	1.4873	2.5428	2.9121	2.6661	2.8726	2.7153	2.8846	2.8594	2.8183	2.4931	2.8123	2.9051
3	1.5642	2.6752	2.9008	2.6609	2.8954	2.8413	2.8313	2.8513	2.8179	2.3088	2.7547	2.8882
4	1.7976	2.6279	2.9028	2.6517	2.8873	2.7437	2.8780	2.8416	2.8779	2.3911	2.7961	2.8702
5	2.6360	2.4077	2.8646	2.7460	2.8905	2.2435	2.9076	2.8856	2.7256	2.7750	2.8147	2.8545
6	2.3191	2.3126	2.8680	2.6015	2.8450	2.1449	2.7264	2.7789	2.8675	2.3156	2.6388	2.8118
7	2.4530	2.6766	2.9243	2.7039	2.8622	2.8455	2.8835	2.8528	2.8922	2.5723	2.8713	2.8924
8	2.4212	2.5125	2.8775	2.6465	2.8054	2.7307	2.8569	2.7969	2.8903	2.7093	2.8188	2.9009
9	2.2075	2.5598	2.9236	2.6181	2.8567	2.8423	2.8830	2.8545	2.7954	2.2688	2.8567	2.8252
10	2.1895	2.5205	2.8712	2.6471	2.8925	2.3005	2.8860	2.8649	2.6801	2.5750	2.8102	2.8892
* "Green color → Red color" equalizes "Best result → Worst result"										0	1.5	3

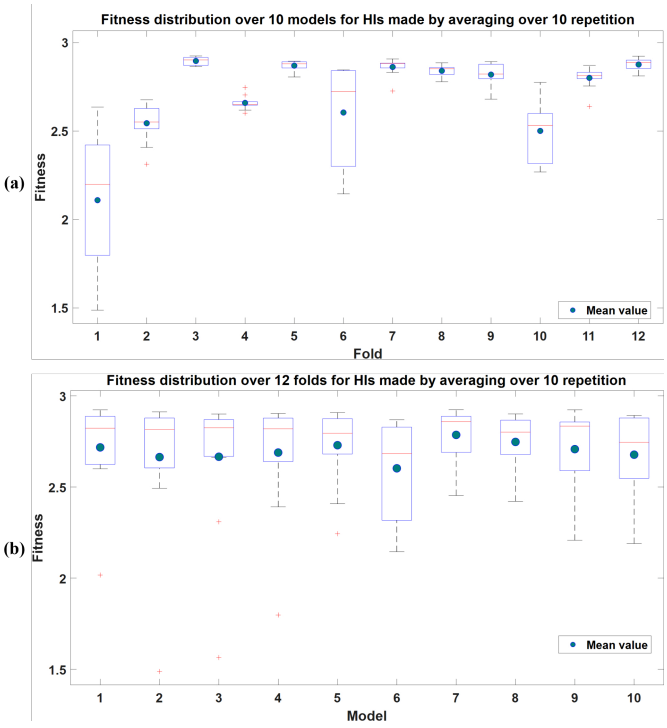


Figure 5.12: The distribution of the Fitness value of ensemble HIs by SAE (across all replications) for various (a) folds and (b) models.

Model 7 has obviously not learned the other SSPs in the training dataset, let alone the test one, SSP 1. It is possible that this is due to inappropriate training progress adjustments for this fold (e.g. validation check patience set to 6), demonstrating the limitations of the proposed DNN models, which can be improved in future frameworks aimed at developing more generalized models and training progress for all folds. Nonetheless, Figure 5.13(b) indicates that all units have comparable patterns to some extent, resulting in a fair Fitness value (2.453). With this in mind, while model 7 could not create HIs following the targets for the first fold (i.e., high RMSE), it could intelligently fuse the input features to produce an average HI that relatively matched the prognostic criteria (i.e., high Fitness), including

Mo (almost increasing), Tr (almost same pattern), and Pr (almost from 0.1 to 0.4). In other words, the model could have discovered how to relate and fuse the features to create a HI with relatively the same pattern for all units.

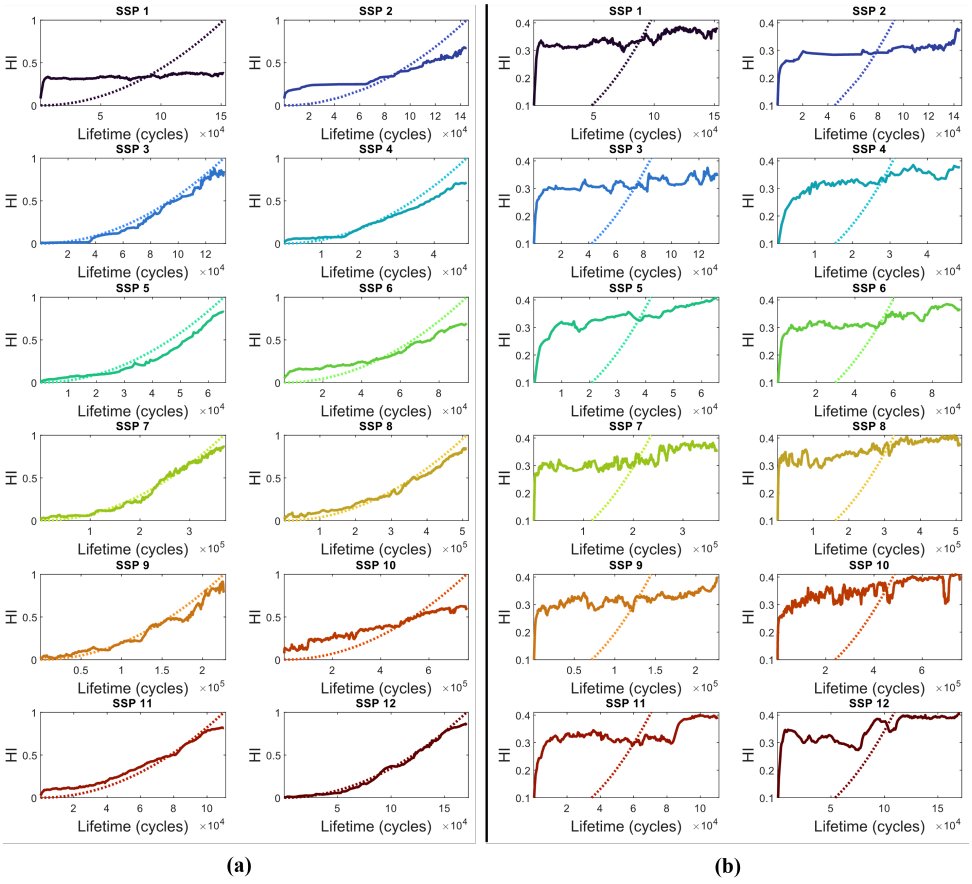


Figure 5.13: Ensemble HIs by SAE across all replications constructed by model 7 for (a) all folds (all units are test datasets obtained from the relevant folds) and (b) fold 1 (only unit 1 is the test dataset (averaged), and the rest are the training ones (averaged)). Dot lines are the ideal HIs.

5.5 2^{nd} framework: FFT-PCA-2S-ML-PBO

The current section presents an ML approach that combines a dimension reduction (PCA-based technique), a historical- or time-independent (TIM), and a historical- or time-dependent (TDM) sub-model after up-sampling the time series in each batch. The overall framework, as depicted in Figure 5.14, encompasses the entire process, from raw AE data to the final HI (the 2^{nd} level HI), including the new proposed approach different from the 1^{st} framework (stages f to j). In addition, after stage j, EL techniques are applied.

The steps of the framework, from adaptive standardization (stage f) through TDM (stage j), establish the primary contributions of the 2^{nd} framework. The steps of adap-

tive standardization (stage f) and dimension reduction using four PCA-based algorithms (stage g) were already explained in Chapter 4. The remaining steps of the framework will be described in this section. It is important to mention that the units (SSPs) are divided into three portions: training (D), validation (v), and test (τ) sets. Additionally, the training data of D itself is further split into training (D') and validation (v') subsets specifically for TIM. This division strategy ensures that TIM is adequately trained and validated before its application to the overall HI construction process.

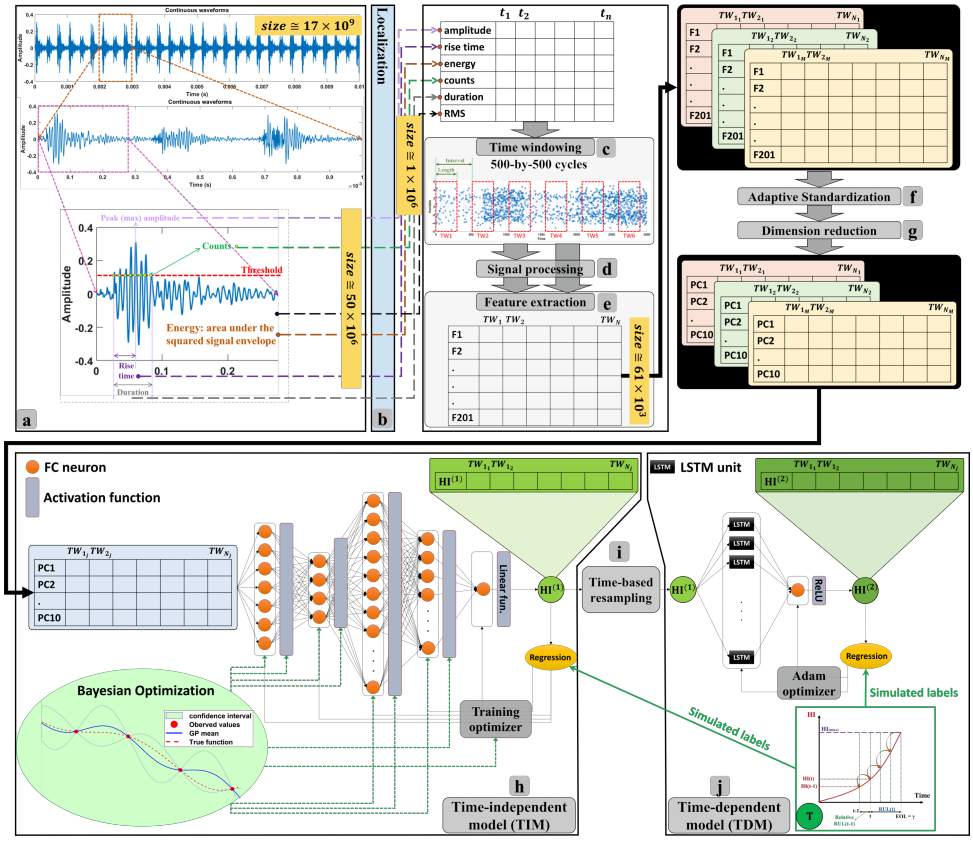


Figure 5.14: Workflow of the 2^{nd} framework developed upon AE data: (a) pre-processing (initial feature extraction), (b) localization, (c) windowing, (d) signal processing, (e) feature extraction, (f) adaptive standardization, (g) dimension reduction, (h) time-independent model, (i) time-based resampling, and (e) time-dependent model

5.5.1 Feature fusion: two-stage ML approach with physics-based Bayesian optimization (2S-ML-PBO)

Constructed on the foundation of the SSL paradigm, the focal concept detailed in Chapter 3, a two-stage ANN architecture named 2S-ML-PBO has been developed. This architecture fundamentally incorporates three components: TIM, time-based resampling, and TDM.

Time-independent model (TIM)

TIM, a historical- or time-independent model, is exclusively focused on the extraction of spatial information. In this context, an initial neural network architecture is established through a trial-and-error process. Following this, the PBO algorithm is employed to optimize the relevant constructive hyperparameters and network architecture, as illustrated in Figure 5.15. Given the dimension reduction resulting in only ten features (PCs), a DNN with a few layers serves as a suitable starting point for the optimizable ANN designated for the regression task. This involves fitting the 10 PCs to a corresponding value (HI). Detailed information about the initial network, including its loss function, hyperparameters, and more, as well as the PBO and its objective function, is provided in the following.

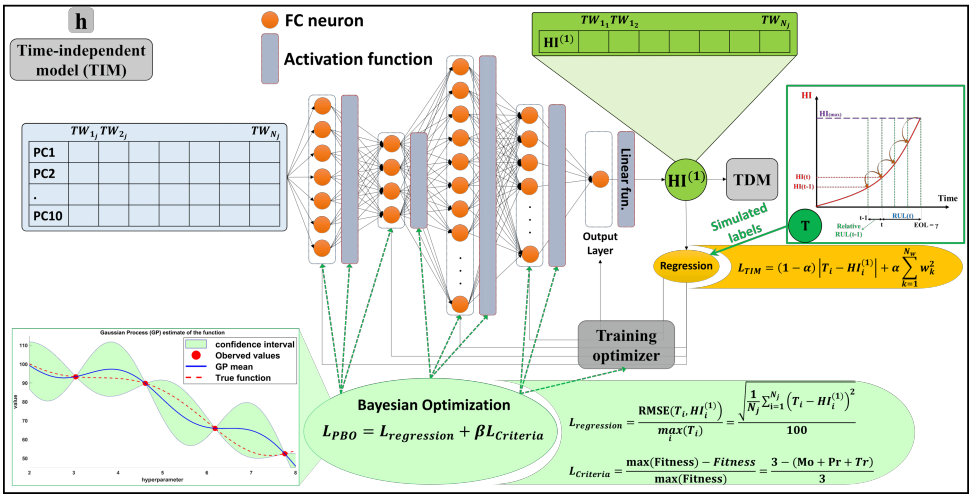


Figure 5.15: Time-independent model (TIM) fed by ten principal components (PCs) and yielding the 1^{st} level HI.

Optimizable neural network architecture of TIM:

A DNN with four layers, including a linear transfer function as the output layer, is designed to fit the 10 PCs to the ideal simulated HI. The loss function used for comparing predictions and targets is a modified mean absolute error (MMAE), given by Eq. 5.1:

$$L_{TIM} = (1-\alpha)\frac{1}{R}\sum_{i=1}^{R}\left|T_i - HI_i^{(1)}\right| + \alpha\sum_{k=1}^{N_w} w_k^2 \quad , \quad i \in D' \tag{5.1}$$

where R represents the number of responses chosen among the training data points ($R \subset [1,\ N_1] \cup [1,\ N_2] \cup \ldots \left[1,\ N_j\right] \ldots \cup [1,\ N_M]$), N_j representing the number of responses for the j^{th} unit and M representing the number of units). T_i and $HI_i^{(1)}$ denote target value and the network's output for response i, respectively. N_w denotes the number of learnable weights of TIM. The regularization parameter α is introduced to enhance generalization by modifying the performance function. The use of this performance function reinforces the NN to have smaller weights and biases, resulting in smoother responses and reducing overfitting. For the current framework, α is set to 1.

The DNN model is trained and validated using only the training units ($i \in D$). 30% of the training data is reserved for validation purposes ($v' \subset D$). While the maximum number of training epochs is set to 1000, the DNN's output is based on the best validation loss, with the validation check patience set to 10. According to the optimizers and their default values in the MATLAB R2022a framework, the remaining hyperparameters are determined through the utilization of the PBO algorithm, which will be further elaborated on in the following subsection.

Physics-based Bayesian optimization (PBO):

The hyperparameters optimized by the BO algorithm include training optimizer algorithms, the number of neurons in each FC layer, and the activation function for all hidden layers. The TIM's weights and biases can be trained using a variety of optimizers. The three types of optimizer algorithms Levenberg-Marquardt (LM) [9–11], Bayesian regularization (BR) [12, 13], and resilient backpropagation (RB) [11, 14] are considered as the search space of the first optimizable variable that can be tuned by BO. The number of neurons allocated for FC layers 1, 2, 3, and 4 are within the ranges [1,50], [1,50], [1,50], and [1,10], respectively. The activation function is the last optimizable variable, and it is assigned the same type for all hidden layers. It is selected from a categorical space according to MATLAB terminology. The aforementioned hyperparameters and their respective search spaces are provided in Table 5.5.

A new objective function inspired by the physics of the problem and prognostics for the BO algorithm is introduced, which considers the evaluation metrics of the HI. Accordingly, the BO algorithm will hereinafter be referred to as "Physics-based BO (PBO)". The PBO objective function comprises two components: regression and criteria loss functions. The regression loss function is based on RMSE between the SS-based simulated labels and predictions, calculated solely on the validation (composite panels) set ($i \in v$). The criteria loss function includes Mo, Pr, and Tr, which are computed using all datasets, including both training and validation portions ($i \in v \cup D$). Since the BO algorithm does not initially have a clue about the optimal solutions' space, this can lead to a wide range of responses. It results in slower convergence and increased time consumption. The regression loss function serves as a guide for the model by providing a general pattern. On the other hand, the criteria loss function strengthens the significance of prognostic metrics to prevent the algorithm from merely adhering to the simulated labels, allowing it to explore other viable and meaningful solutions. This strategy drives the BO algorithm to seek out diverse and potentially superior solutions, improving its capacity to adapt and converge towards better outcomes.

The equations of the regression and criteria loss functions that constitute the PBO objective function are as follows:

$$L_{regression} = \frac{RMSE\left(T_i, HI_i^{(1)}\right)}{\max_i(T_i)} = \frac{\sqrt{\frac{1}{N_j}\sum_{i=1}^{N_j}\left(T_i - HI_i^{(1)}\right)^2}}{100}, \quad i \in v, \quad j \in M \tag{5.2}$$

$$L_{Criteria} = \frac{\max(Fitness) - Fitness}{\max(Fitness)} = \frac{3 - (Mo + Pr + Tr)}{3}, \quad i \in (v \cup D) \tag{5.3}$$

$$L_{PBO} = L_{regression} + \beta L_{Criteria} \tag{5.4}$$

Table 5.5: The hyperparameters of TIM and their search spaces for optimization by PBO.

Hyperparameter	Search space		Type
Optimizer	Levenberg-Marquardt (LM) Bayesian regularization (BR) Resilient backpropagation (RB)		categorical
Number of neurons at FC layer 1	[1,50]		integer
Number of neurons at FC layer 2	[1,50]		integer
Number of neurons at FC layer 3	[1,50]		integer
Number of neurons at FC layer 4	[1,10]		integer
Activation functions	***Name***	***function***	categorical
	Linear	$\sigma(z) = z$	
	Rectified linear units (ReLU)	$\sigma(z) = \begin{cases} 0 & : z \le 0 \\ z & : z > 0 \end{cases}$	
	Saturating linear	$\sigma(z) = \begin{cases} 0 & : z \le 0 \\ z & : 0 \le z \le 1 \\ 1 & : z \ge 1 \end{cases}$	
	Symmetric saturating linear	$\sigma(z) = \begin{cases} -1 & : z \le -1 \\ z & : -1 \le z \le 1 \\ 1 & : z \ge 1 \end{cases}$	
	Hard-limit	$\sigma(z) = \begin{cases} 0 & : z < 0 \\ 1 & : z \ge 0 \end{cases}$	
	Symmetric hard-limit	$\sigma(z) = \begin{cases} -1 & : z < 0 \\ 1 & : z \ge 0 \end{cases}$	
	Log-sigmoid	$\sigma(z) = \frac{1}{1+\exp(-z)}$	
	Hyperbolic tangent sigmoid	$\sigma(z) = \frac{2}{1+\exp(-2z)} - 1$	
	Elliot symmetric sigmoid	$\sigma(z) = \frac{z}{1+\lvert z\rvert}$	
	Radial basis	$\sigma(z) = \exp(-z^2)$	
	Normalized radial basis	$\sigma(z) = \frac{\exp(-z_i^2)}{\sum_{j=1}^{J}\exp(-z_j^2)}, \quad i = 1, 2, 3, \ldots, J$	
	Softmax	$\sigma(z) = \frac{\exp(z_i)}{\sum_{j=1}^{J}\exp(z_j)}, \quad i = 1, 2, 3, \ldots, J$	
	Triangular basis	$\sigma(z) = \begin{cases} 0 & : z < -1 \\ 1 - \lvert z\rvert & : -1 \le z \le 1 \\ 0 & : z > 1 \end{cases}$	
	Inverse	$\sigma(z) = \frac{1}{z}$	
	Competitive	$\sigma(z) = \begin{cases} 1 & : j = \arg\max_i(z_i), \quad i = 1, \ldots, J \\ 0 & : j \ne \arg\max_i(z_i), \quad i = 1, \ldots, J \end{cases}$	

The parameter β determines the significance of $L_{Criteria}$ in relation to $L_{regression}$. $L_{Criteria}$ is normalized using the maximum fitness score, resulting in a range of [0, 1]. On the other hand, $L_{regression}$ is normalized based on the maximum target value, which is 100. It should be noted that the ideal HI values are simulated within a range from 0 (representing a healthy state) to 100 (indicating a failure state) for this framework. The PBO algorithm was given 100 trials with an exploration ratio of 0.8 in parallel computing to optimize the

hyperparameters.

Time-based resampling

After the TIM step, the 1^{st} level predicted HI can serve as a prognostic parameter to be imported into a prognostic model for predicting RUL. Despite the undeniable historical dependence in SHM data, rooted in the physics of fatigue, progressive damage, and the passive nature of the AE technique, this aspect has been overlooked in the current framework. Therefore, a historical- or time-dependent model, TDM, should be designed to extract the temporal information. Prior to designing the TDM, the input data needs to undergo resampling to ensure that all sequences of input $HI^{(1)}$ within a batch have the same length. Conventional padding techniques like zero padding are unsuitable in this context, as the HI values with respect to the percentage of lifetime should exhibit similarities. For example, if the batch size is 2 and the lengths of the two intended HIs are 100 and 1000, extending the first HI with 900 zero values to match the length of the second HI would incorrectly set the HI at the EoL for the first unit as 0 instead of its actual (compatible) value, which should be 100, as in the second HI. Likewise, conventional interpolation methods are not viable due to the non-linear correlation between the number of data points in HI and the EoL. Due to the varying recording rate of the passive AE system depending on the predetermined amplitude threshold value and uncertain progressive damage, the length (number of data points) of HI for a unit with a bigger EoL may be shorter than that with a smaller EoL.

To address these challenges, the current study adopts a technique for up-sampling referred to as "time-based resampling" (see Figure 5.16). The first step is to convert the time vectors of HIs into percent lifespans, with values in the range of [0%, 100%]. The shorter HI vectors (in terms of the quantity of data points) in each batch are then up-sampled (based on the associated time vectors) to match the length of the longest HI vector. For every batch separately, this procedure is performed. It is crucial to remember that in such a scenario, the batch size cannot be the same as the total number of training datasets. This is due to the TIM's tendency to just learn the position of data, disregarding its value, leading it to predict from 0 (healthy) at the beginning to 100 (failure) at the EoL based exclusively on the position of the incoming data. For instance, if HIs of equal lengths of 1000 are employed, the TIM model would learn that, regardless of the input value, position 1 should result in a zero value and position 1000 should result in a hundred. As the test set should not logically be resampled, training a model in this way would result in much worse performance on the test set. As a result, the batch size should be less than all of the training data. In the current investigation, 10 units are kept for training, and one panel is designated for testing and another for validation in all scenarios. Consequently, a batch size of 2 or 5 appears more reasonable, with 2 being the preferred choice. Lastly, it should be noted that if a batch size of 1 is chosen, time-based resampling becomes irrelevant.

Time-dependent model (TDM)

In this subsection, a seq2seq regression model called TDM is introduced to address the historical dependence (HD) among AE data. Prior to this, using the previous regression model, TIM, 10 high-level extracted features are mapped to the 1^{st} level HI without considering the HD among AE data. The TDM model takes into account the temporal relationship between data points, which is crucial for accurate prognostic applications.

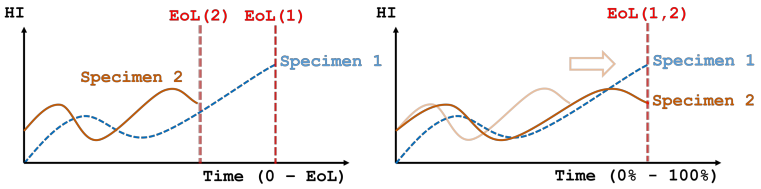


Figure 5.16: Sequence length synchronization and time-based resampling.

To maintain a long-term record of sequential inputs, the LSTM layer [15] is a suitable candidate and serves as a key component in the TDM (see Figure 5.17). The LSTM layer consists of 10 units, initialized with zero values for hidden and cell states. The pseudo-Huber loss function, which is a smooth approximation of the Huber loss function [16], is used for the seq2seq regression task [14, 17]:

$$L_{TDM} = \frac{1}{N_j}\sum_{i=1}^{N_j}\delta^2\left(\sqrt{1+\left(\frac{T_i - HI_i^{(2)}}{\delta}\right)^2}-1\right), \quad i \in D\,, \quad j \in M \tag{5.5}$$

Here, δ controls the steepness at extreme values and is set to 20 in our work after trial and error (20% of the maximum target value, considering that $\max_i(T_i) = 100$). An Adam optimizer [8] is employed to train the TDM, with an initial learning rate of 0.01, a learning rate drop factor of 0.1, a learning rate drop period of 10, and a gradient threshold of 1, all determined through trial and error. The network's output is based on the best validation loss ($j \in v$), with the validation check frequency set to 50 iterations (the number of learned batches) and the validation check patience set to 50, despite the fact that the maximum number of training epochs was set to 2000. Since the sequences in each batch are already identical in length, a batch size of 2 was chosen, as described in the preceding section.

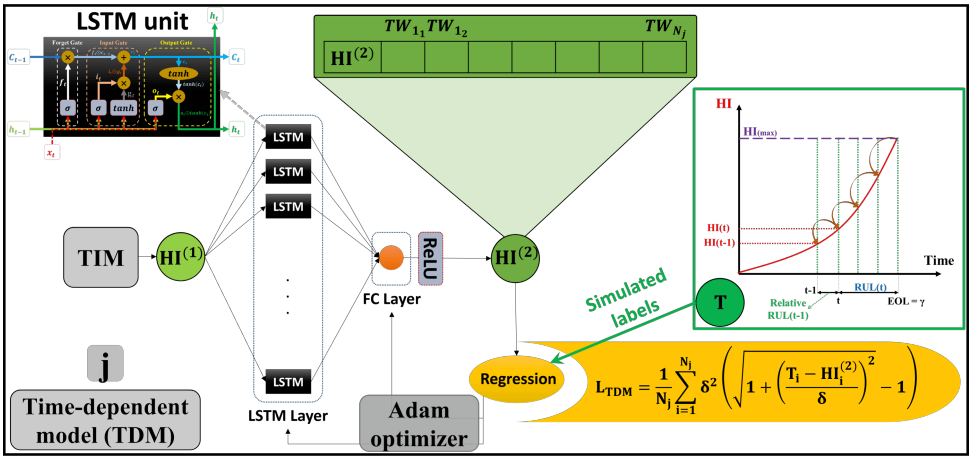


Figure 5.17: Time-dependent model (TDM) fed by the 1^{st} level HI and yielding the 2^{nd} level HI.

Ensemble learning (EL)

After building HIs with TIM-TDM models, EL techniques can address randomness and uncertainties by creating a meta-model. While exploring several EL techniques, the second framework employs averaging EL techniques with distinct weighting methods discussed in the previous chapter. Unlike the fixed architecture of TDM, it is important to highlight that the TIM component of the base models, involving architecture optimization through PBO, dictates whether the base learners are homogeneous or heterogeneous. When TIM is executed individually for each validation index-based combination, the resulting TIM architectures, optimizers, and other hyperparameters are likely to differ. However, to mitigate resource-intensive processes, TIM's hyperparameters are optimized using PBO solely for the first validation index-based combination in the set ($v \cup D$). For the remaining combinations, the same hyperparameters are retained, resulting in homogeneous TIMs across the 110 base learners.

5.5.2 Results

To comprehensively assess various combinations and ascertain the effectiveness, validity, and stability of the proposed framework, all 12 folds were examined. Within each fold, a single stiffener composite panel was designated as the test set, while the remaining panels served as training and validation data. Since there are 11 alternatives that can be used as validation and this choice is effective, this process was repeated across 11 different validation index-based combinations, iterated 10 times using distinct random seed numbers. Essentially, this approach emulates the LOOCV methodology, wherein a holdout validation strategy is adopted within each fold. This strategy affirms the evaluation of the model's generalizability. These 1320 runs ($12 \times 11 \times 10$) were conducted for all four types of PCA-based data reduction techniques to facilitate a comprehensive comparison.

In this section, the results of the proposed methodology up to the ensemble stage, termed base learner models, are first presented. Subsequently, the outcomes of the ensemble learner models are explored.

Base learner models

Figure 5.18 displays the simulated ideal HIs alongside a selection of promising candidates from the outcomes of $HI^{(1)}$s (TIM outputs) and $HI^{(2)}$s (TDM outputs) for different PCA-based techniques. Each case represents a holdout validation scenario. The error depicted in Figure 5.18 illustrates the RMSE between the simulated ideal HIs and the constructed HIs. Notably, $HI^{(1)}$s generated by TIM exhibit significant fluctuations, whereas TDM yields smoother $HI^{(2)}$s. There have been some noteworthy observations, highlighting the limitations of TIM, such as cases where $HI^{(1)}$s exhibit a declining trend. On the other hand, TDM successfully corrects such patterns. Concerning the behavior of $HI^{(2)}$s, multiple incremental steps were observed over the fatigue life in many instances (e.g., A.1 in Figure 5.18. These steps can potentially signify distinct damage states, offering insights for subsequent prognostic models in RUL prediction. However, linking these steps with physical damage states in a stable and meaningful manner necessitates substantial effort and future experiments.

Table 5.6 presents the evaluation metrics for the HIs shown in Figure 5.18. Notably, due to its HD nature, all scores for $HI^{(2)}$s surpass those for $HI^{(1)}$s. The proposed model not

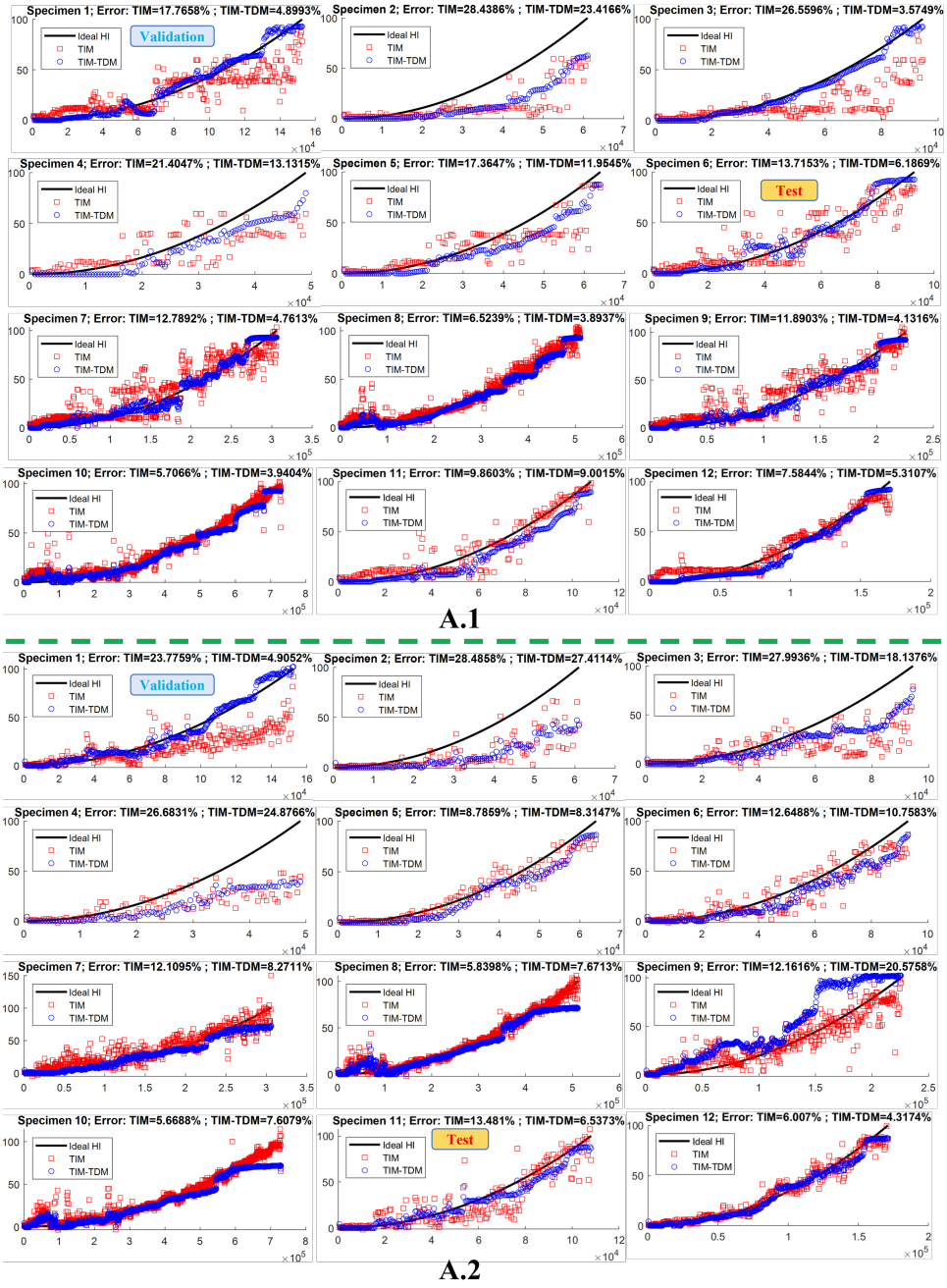


Figure 5.18: Qualified candidates for $HI^{(1)}$s (TIM outputs) and $HI^{(2)}$s (TDM outputs) for different PCA-based techniques (A.1, A.2, B.1, and B.2) - continued on the next page

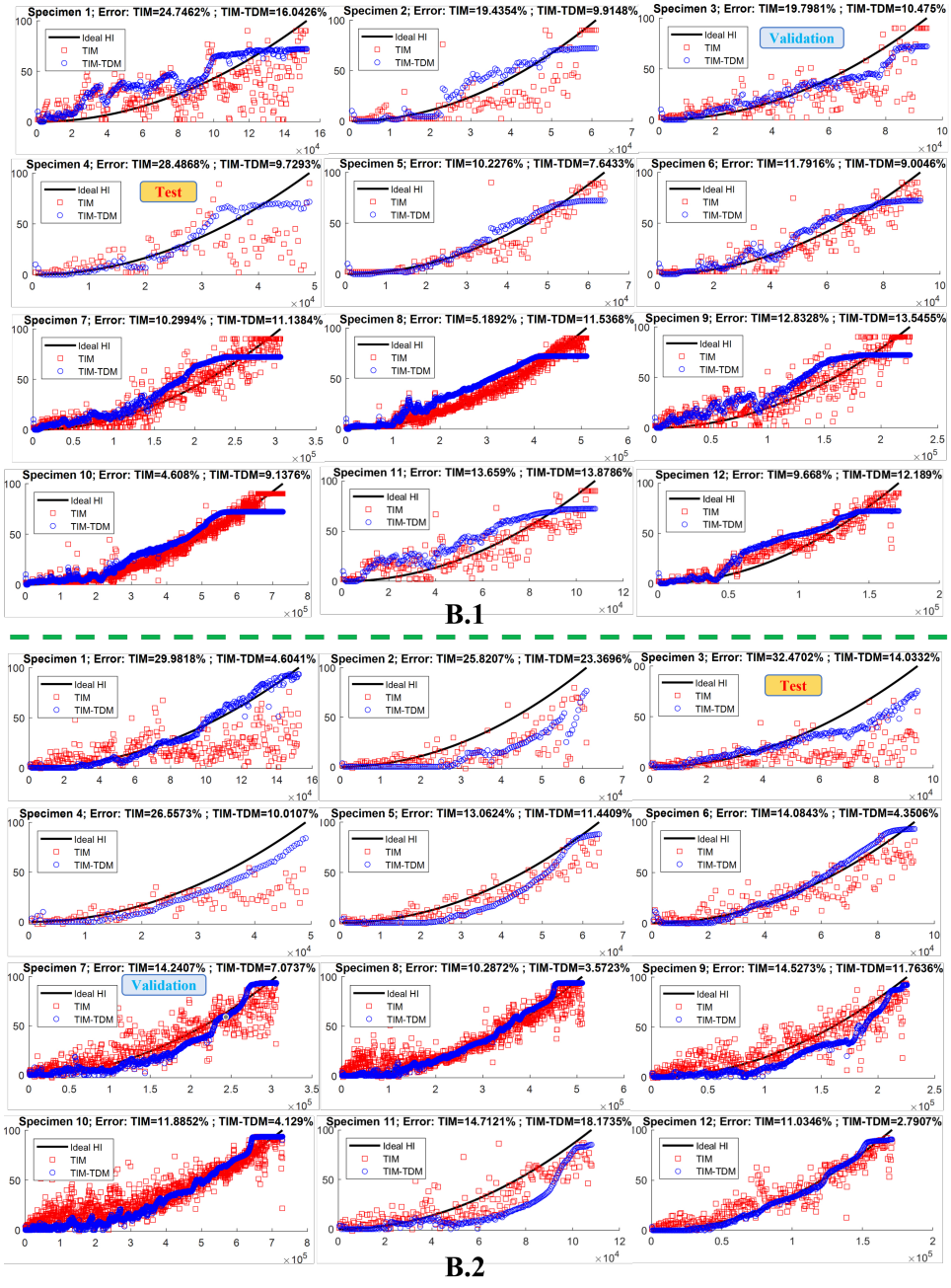


Figure 5.18: Qualified candidates for $HI^{(1)}$s (TIM outputs) and $HI^{(2)}$s (TDM outputs) for different PCA-based techniques (A.1, A.2, B.1, and B.2) - continued from the previous page

only provides a streamlined and faster approach but also yields elevated Fitness scores in comparison to the 1^{st} framework. Worth highlighting is that the DL model in the 1^{st} framework incorporates 193,418 learnable parameters, while the proposed method employs only between 662 and 4,924 learnable parameters, averaging around 1,731 (approximately 0.9%). Out of this total, 491 parameters are allocated to TDM, and the quantity varies for TIM depending on its optimized architecture.

Table 5.6: The desirable candidate outcomes of each model run individually for various PCA methods.

PCA	Test	Validation	Mo		Pr		Tr		Fitness		RMSE of test		Learnable
type	specimen	specimen	$HI^{(1)}$	$HI^{(2)}$	$HI^{(1)}$	$HI^{(2)}$	$HI^{(1)}$	$HI^{(2)}$	$HI^{(1)}$	$HI^{(2)}$	$HI^{(1)}$	$HI^{(2)}$	parameters
A.1	6	1	0.98	1.00	0.82	0.90	0.66	0.94	2.46	2.84	13.72	6.19	704
A.2	11	1	0.99	1.00	0.72	0.76	0.63	0.92	2.34	2.68	13.48	6.54	1217
B.1	4	3	0.98	1.00	0.92	1.00	0.43	0.91	2.32	2.90	28.49	9.73	1662
B.2	3	7	0.95	1.00	0.71	0.92	0.40	0.90	2.06	2.82	32.47	14.03	2794

Furthermore, it is essential to emphasize that in the 1^{st} framework, the LOOCV procedure employed 11 units for training and 1 unit for validation. That approach yielded results based on the network's output with the best validation loss, which was also the same as the test unit. In contrast, in the 2^{nd} framework, the LOOCV process employs different units for validation and testing, thus achieving a higher degree of generalization.

The comprehensive outcomes of TIM-TDM across various subsets and PCA-based techniques are depicted in Figures 5.19 and 5.20, illustrating the RMSE and Fitness scores, respectively. The results indicate the mean values (displayed on the right y-axis) over ten replications (shown on the x-axis) for each subset (validation index-based combinations displayed on the left y-axis), with error (standard deviation) bars.

Figure 5.19 shows the RMSE between the simulated ideal HIs and the constructed HIs by the developed model ($HI^{(2)}$). With the exception of specimen 9, the B.1 PCA-based technique generally yields a lower average RMSE compared to others, notably evident in specimens 2, 4, and 5. Following B.1, A.2 (observed in specimens 5 and 10) and B.2 (seen in specimens 8 and 9) demonstrate better RMSE results on average. Yet, it is important to note that RMSE outcomes vary based first on the test unit and then on the validation unit. For instance, B.1 is consistently the best choice for test specimen 5, regardless of the chosen validation specimen. However, this trend might not hold true for test specimen 6, as its superiority depends on the selection of the validation specimen.

Given that the primary objective is to provide qualified HIs based on prognostic criteria, Figure 5.20 holds greater significance than Figure 5.19. The outcomes presented in Figure 5.20, showcasing Fitness scores, exhibit a higher degree of stability compared to the RMSE results in Figure 5.19. Across various units (except for 2, 3, and 8), the B.1 PCA-based technique consistently yields higher Fitness scores, while the A.2 PCA version displays the lowest average Fitness score. The test unit that poses the greatest challenge for acceptable PCA versions is 5, where the discrepancy with the impractical version B.1 is substantial. Unit 8 presents challenges across all PCA versions. Notably, in the exceptional case of unit 3, the B.2 PCA version stands out as the most effective. The key point is that the choice of a suitable validation unit can significantly influence the score. For instance, in test unit 11 with the best validation unit 1, the score becomes highly acceptable, even for the A.2 PCA version.

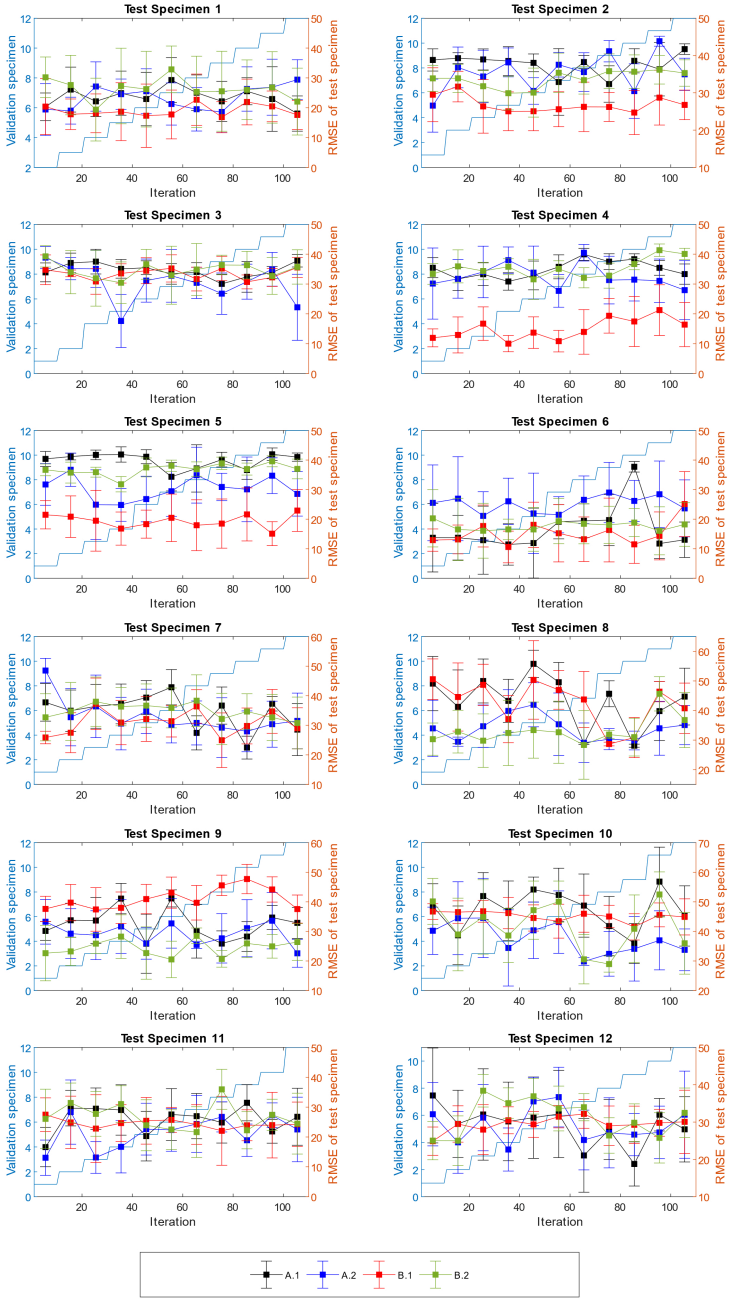


Figure 5.19: RMSE between ideal HIs and constructed HIs by the 2^{nd} framework over various subsets (train, validation, and test combinations) and PCA-based techniques (A.1, A.2, B.1, and B.2)

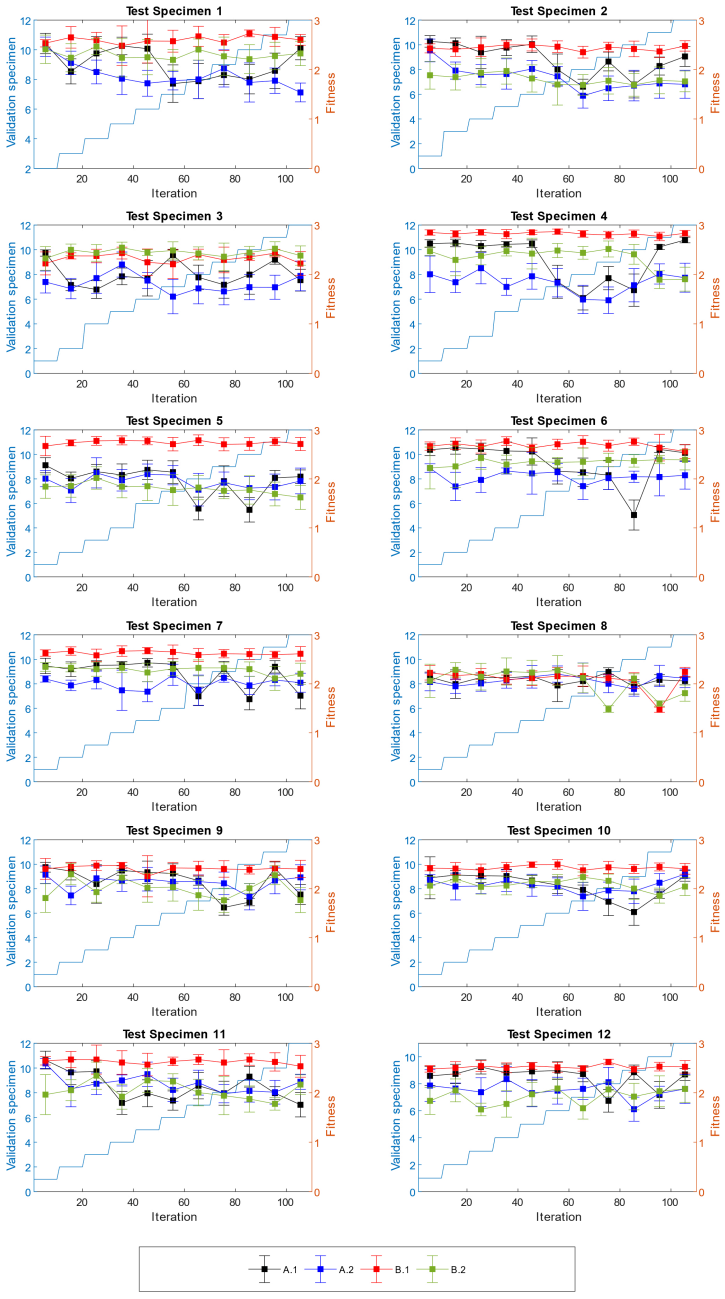


Figure 5.20: Fitness for the 2^{nd} framework over various subsets (train, validation, and test combinations) and PCA-based techniques (A.1, A.2, B.1, and B.2)

Ensemble learner models

In this subsection, the outcomes of ELMs are presented, covering both SAE and WAE. Various ω_k values are explored for WAE, encompassing ω_k^{MSE}, $\omega_k^{Fitness}$, and $\omega_k^{Fitness\_MSE}$ as defined in Eqs. 4.45, 4.46, and 4.47, respectively. For $\omega_k^{Fitness}$, various sets of coefficients $\{a, b, c\}$ are considered, including $\{1, 1, 1\}$, $\{1, 1, 0\}$, $\{1, 0, 0\}$, and $\{1, 0.5, 0.25\}$, to assess the impact of diverse prognostic metrics. As for $\omega_k^{Fitness\_MSE}$, the coefficient set $\{1, 1, 1\}$ is exclusively taken.

Figure 5.21 depicts the distributions of the Fitness score across 110 subsets (resulting from 11 validation-index-based combinations × 10 replications) for each fold. The x-axis represents the fold, or, in other words, the single-stiffener composite panel chosen as the test unit for the model. This figure offers insights into the various versions of PCA and ELMs employed. Additionally, Appendices (A.2) contains figures showcasing all the constructed HIs post-EL for different versions of PCA.

As observed in Figure 5.21, the EL step, regardless of its type, consistently enhances the score in comparison to the mean value across all cases. This improvement is particularly notable for the A.2 PCA version, while it has a relatively smaller impact on the B.1 PCA version. Specifically, the B.1 PCA version yields a more consistent distribution of highly qualified HIs with less variance, as indicated by the box plot. However, it is crucial to note that this PCA version is unsuitable for prognostics, as previously discussed.

Certain folds present greater challenges for different PCA versions. For instance, specimens 5, 3, and 1 are more challenging for the A.1 PCA version, while specimens 3, 2, 4, and 1 pose challenges for the A.2 PCA version. The B.1 PCA version encounters difficulties with specimens 8 and 3, while the B.2 PCA version struggles with specimens 2 and 5.

While examining various applied EL techniques, it is notable that WAE-MSE does not necessarily outperform SAE. However, WAE-Fitness and WAE-FitnessMSE consistently yield better outcomes than SAE. Focusing on WAE-Fitness, it is evident that the prognostic metrics can sometimes oppose each other. For instance, taking composite panel 6 for the A.2 PCA version (refer to Figure A.7), considering the metrics coefficient set $\{1, 1, 0\}$ achieves the desired Pr, an important aspect. Conversely, when the Tr coefficient increases (in cases of $\{1, 1, 1\}$ or $\{1, 0.5, 0.25\}$), Pr decreases. When Pr is achieved, the challenge of extrapolation in RUL prediction can be transformed into an interpolation problem, and as is well known, an extrapolation problem still poses a significant challenge in the field of ML.

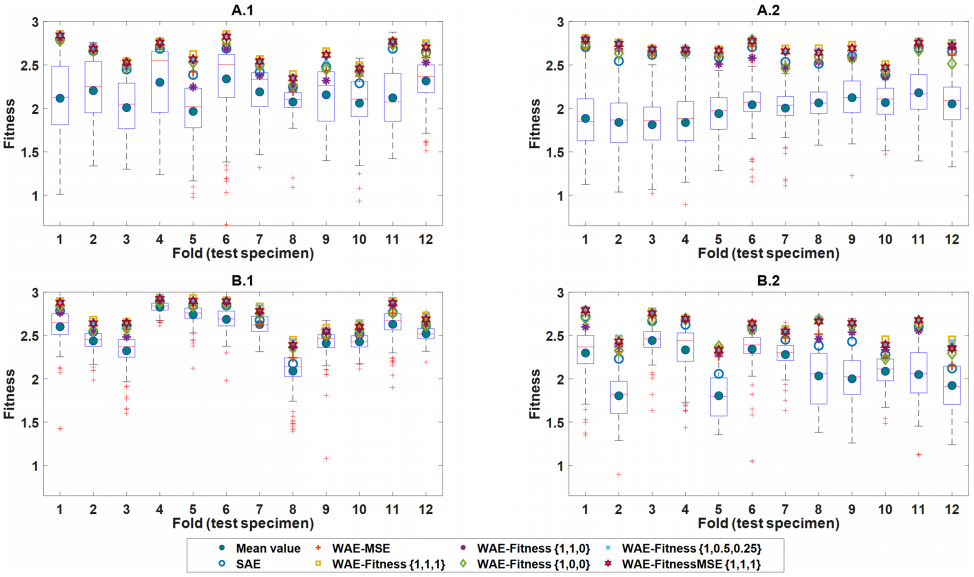


Figure 5.21: Distribution of Fitness for the 2^{nd} framework over various subsets (train, validation, and test combinations) and versions of PCA (A.1, A.2, B.1, and B.2). The box plot and the mean value (represented by a dark blue-filled circle) pertain to results before EL, while the other markers depict the scores of the HIs after EL.

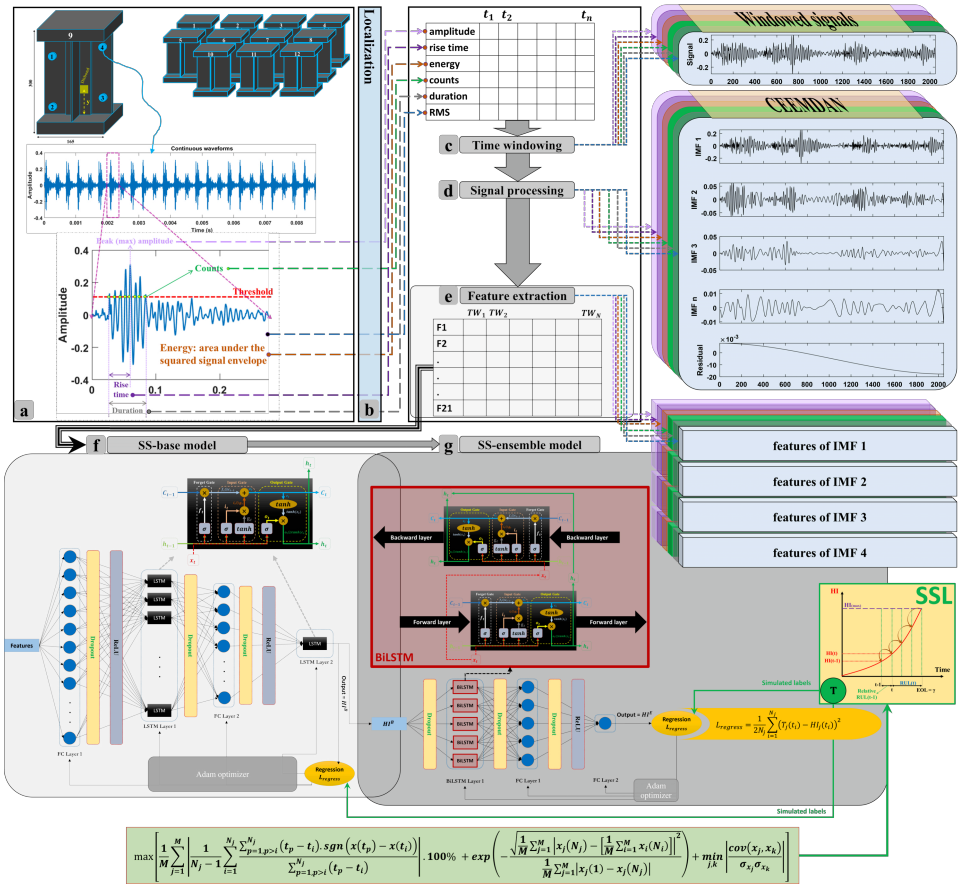
5.6 3^{rd} framework: CEEMDAN-SSEDL

The 3^{rd} new AI-based approach, as illustrated in Figure 5.22, incorporates CEEMDAN for feature extraction and semi-supervised base deep learner models made of LSTM layers for information fusion. Ensemble learning, especially using a SS network built with BiLSTM, improves HI quality while reducing deep learning randomness.

Feature extraction: As mentioned in Section, six low-level features are obtained from each AE event, i.e., A, R, E, CNTS, D and RMS. For the i^{th} unit, there are k_i time windows. The CEEMDAN methodology is applied to each time window in order to extract the IMFs. Arbitrarily, four IMFs are extracted for each time window. Since the time windows have different numbers of data points and occasionally not enough data is available to decompose into IMFs, linear interpolation between the previous and next time windows is performed.

For each low-level feature, four IMFs are extracted, providing 24 new features. For each time window, 21 statistical quantities are calculated, which are listed in Table 5.7. In total, 504 (6x4x21) new features are obtained, which are going to be the input to the DL algorithm that creates the HI.

Standardization (zero-mean): Using a zero-mean normalization technique (Chapter 4) that solely uses the mean value and standard deviation of the training dataset, the network's input features are standardized.



$$\max\left[\left|\frac{1}{M}\sum_{j=1}^{M}\left|\frac{1}{N_j-1}\sum_{t=1}^{N_j}\frac{\sum_{p=1,p>t}^{N_j}(t_p-t_i).sgn\left(x(t_p)-x(t_i)\right)}{\sum_{p=1,p>t}^{N_j}(t_p-t_i)}\right|.100\% + exp\left(-\frac{\sqrt{\frac{1}{M}\sum_{j=1}^{M}\left|x_j(N_j)-\left[\frac{1}{M}\sum_{i=1}^{M}x_i(N_i)\right]\right|^2}}{\frac{1}{M}\sum_{j=1}^{M}\left|x_j(1)-x_j(N_j)\right|}\right) + \min_{j,k}\left|\frac{cov(x_j,x_k)}{\sigma_{x_j}\sigma_{x_k}}\right|\right]$$

Figure 5.22: The overall 3^{rd} proposed framework: (a) AE monitoring and low-level feature extraction; (b) localization; (c) windowing; (d) signal processing (CEEMDAN); (e) statistical feature extraction; (f) semi-supervised base learner model; (g) semi-supervised ensemble learner model.

Table 5.7: Statistical features extracted from IMFs.

No.	Name	No.	Name	No.	Name
1	Mean	8	Kurtosis	15	Central moment for 4th order
2	Standard Deviation	9	Crest Factor	16	Central moment for 5th order
3	Root Amplitude	10	Clearance factor	17	Central moment for 6th order
4	Root Mean Square	11	Shape factor	18	FM4
5	Root sum of Squares	12	Impulse factor	19	Median value
6	Peak	13	Maximum to minimum difference	20	Signal Power
7	Skewness	14	Central moment for 3rd order	21	Entropy

5.6.1 Feature fusion: semi-supervised ensemble deep learning (SS-EDL)

The same multi-layer LSTM network architecture proposed for the 1^{st} framework (Figure 5.5) is used to perform the feature fusion task in the 3^{rd} framework. The half-mean-squared error is regarded as the loss function of the seq2seq regression network. The same

ten top models (Table 5.2) obtained by BO, given the statistical features extracted from the time and frequency domains in the 1^{st} framework, are considered the base learners in the 3^{rd} framework, where the new features extracted from the IMFs of CEEMDAN are imported. Although the process of optimizing the hyperparameters based on the new feature inputs could be repeated, this step is skipped to save time. It should be noted that this section can be viewed as a type of transfer learning designed to cut down on time.

Ensemble learning (EL)

Once HIs are constructed using base deep learner models, ensemble learner models can handle uncertainties and randomness. Initially, LOOCV is employed, reserving a single unit for testing and utilizing the remaining 11 specimens for subsequent processing. Three main dataset divisions, as explained in Chapter 4, Section 4.6.3, guide this process. For the third framework, Cases B and C are investigated, with Case C involving the random selection of the validation SSP from the remaining 11 SSPs. The base learner models are trained a total of 100 times. In Case C, the validation unit is randomly selected 10 times, and the learning process is conducted with 10 different random seed numbers for initializing weights and biases. Subsequently, the 100 HIs predicted by the base learner models are ensembled using a process involving SAE, WAE, and finally DL models.

Beyond averaging ensemble models, this framework assesses 12 networks with various types of layers, including FC, LSTM, and BiLSTM layers. The architectures (hidden layers) of these EL models are summarized in Table 5.8, with values in parentheses indicating the number of neurons, units, or dropout percentages. An example of the ensemble learner, Model 16– Net(12), is illustrated in Figure 5.23.

Table 5.8: Ensemble learner models used in the 3^{rd} framework.

Model num.	Model name	Architecture (hidden layers)						
1	SAE							
2	WAE-MSE							
3	WAE-RMSE							
4	WAE-Fitness							
5	Net(1)	FC(10)	D(0.5)	ReLU	FC(1)			
6	Net(2)	FC(100)	D(0.5)	ReLU	FC(1)			
7	Net(3)	FC(10)	D(0.5)	ReLU	FC(5)	D(0.5)	ReLU	FC(1)
8	Net(4)	FC(100)	D(0.5)	ReLU	FC(5)	D(0.5)	ReLU	FC(1)
9	Net(5)	LSTM(5)	D(0.5)	FC(5)	D(0.5)	ReLU	FC(1)	
10	Net(6)	LSTM(10)	D(0.5)	FC(5)	D(0.5)	ReLU	FC(1)	
11	Net(7)	FC(10)	D(0.5)	ReLU	LSTM(5)	D(0.5)	ReLU	FC(1)
12	Net(8)	FC(10)	D(0.5)	ReLU	BiLSTM(5)	D(0.5)	ReLU	FC(1)
13	Net(9)	BiLSTM(5)	D(0.5)	BiLSTM(1)	D(0.5)	FC(1)		
14	Net(10)	BiLSTM(10)	D(0.5)	BiLSTM(1)	D(0.5)	FC(1)		
15	Net(11)	BiLSTM(5)	D(0.5)	FC(5)	D(0.5)	ReLU	FC(1)	
16	Net(12)	D(0.5)	BiLSTM(5)	D(0.5)	FC(5)	D(0.5)	ReLU	FC(1)

5.6.2 Results

Base learner models

To demonstrate the effectiveness of the CEEMDAN features as input to the base learner models, the average Fitness values (and standard deviation) over the 100 repetitions are presented in Table 5.9. The best obtained value is 2.82 (±0.24) for the Fold 12 and the

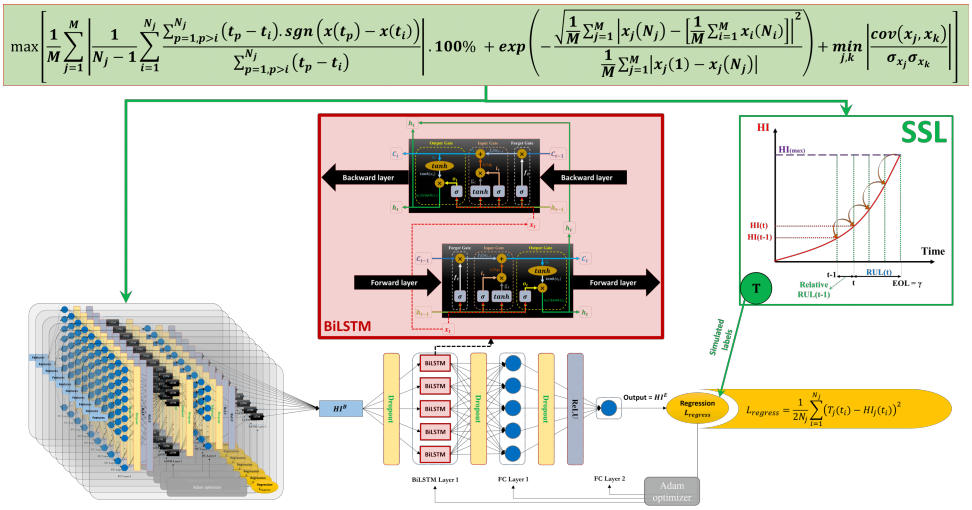


Figure 5.23: The architecture of the semi-supervised ensemble learner Model 16 – Net(12).

base learner Model 10. It is also important to evaluate the prognostic metrics specifically for the test units (Eq. 3.8) to check how well they adhere to these metrics and how well the methodology generalizes to unknown input data. Table 5.10 summarizes these results over the 100 repetitions. The best result is observed for Model 10, Fold 12, with a Fitness 2.75 (±0.21). In Figure 5.24, a visual representation of Tables 5.9 and 5.10 for Model 10 is displayed for easier comparison between the two. What is observed is that Mo and Pr, and consequently Fitness, display slightly lower values. To clarify, the test unit is not able to follow the ideal HI functions with the same proficiency as the training units, and hence, when calculating these prognostic metrics, excluding the training units results in reduced overall values.

A drawback of the original methodology's evaluation via LOOCV, also presented in Ref. [3], is the use of the test unit as validation during training (Case B). This way, it is incorporated into the training step, and consequently, it is not hidden during the application step. Although this provides high Fitness results, its applicability in real-world scenarios is limited due to its dependence on validating using unknown data. To overcome the limitation and enable exploring the possibility of applying the methodology in real time, a random SSP from the training set is used for model validation (Case C). By employing this methodology, the dependency on the test unit is eliminated, thus enabling the possibility of real-time implementation.

The average Fitness over the 100 repetitions for both the entire dataset and the test units are shown in Tables 5.11 and 5.12, while Figure 5.25 presents both cases for the base learner Model 9. In comparison with the methodology using the test unit for early stopping and validation, it is evident that the values obtained are lower, which is expected given that the test unit is now unknown. However, the values remain high, providing HIs with great prognostic potential. The best values are observed for Fold 12 and Model 9 at 2.51 (±0.43) for the entire set and 2.44 (±0.4) for the test unit.

The best HI obtained by Model 10 at iteration 96 and Model 9 at iteration 97 for data

Table 5.9: Fitness values for base learner models of the 3^{rd} framework averaged over the 100 repetitions using Eq. 3.4 and data division Case B.

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Fold 11	Fold 12
Model 1	2.07 (±0.67)	2.21 (±0.61)	1.82 (±0.65)	2.75 (±0.26)	2.58 (±0.48)	2.81 (±0.17)	2.45 (±0.5)	1.98 (±0.35)	2.26 (±0.57)	1.26 (±0.3)	2.48 (±0.48)	2.71 (±0.23)
Model 2	2.35 (±0.6)	2.24 (±0.64)	1.66 (±0.63)	2.74 (±0.26)	2.59 (±0.45)	2.69 (±0.35)	2.49 (±0.44)	2.07 (±0.38)	2.25 (±0.62)	1.34 (±0.35)	2.48 (±0.54)	2.6 (±0.38)
Model 3	2.58 (±0.41)	2.12 (±0.68)	1.67 (±0.61)	2.68 (±0.28)	2.43 (±0.47)	2.65 (±0.31)	2.56 (±0.35)	2.16 (±0.38)	1.97 (±0.71)	1.47 (±0.48)	2.49 (±0.49)	2.54 (±0.33)
Model 4	2.22 (±0.6)	2.09 (±0.66)	1.61 (±0.64)	2.71 (±0.31)	2.5 (±0.45)	2.72 (±0.38)	2.54 (±0.43)	2.1 (±0.37)	2.17 (±0.62)	1.42 (±0.43)	2.32 (±0.58)	2.58 (±0.32)
Model 5	2.56 (±0.65)	1.71 (±0.62)	2.35 (±0.57)	2.7 (±0.27)	2.75 (±0.3)	2.59 (±0.38)	2.48 (±0.46)	1.93 (±0.33)	2.5 (±0.44)	1.32 (±0.27)	2.42 (±0.56)	2.78 (±0.25)
Model 6	2.07 (±0.65)	1.82 (±0.58)	2.1 (±0.55)	2.39 (±0.5)	2.33 (±0.55)	2.67 (±0.26)	2.31 (±0.51)	1.82 (±0.34)	2.06 (±0.61)	1.39 (±0.28)	1.93 (±0.58)	2.51 (±0.39)
Model 7	2.38 (±0.65)	2.12 (±0.7)	1.72 (±0.63)	2.75 (±0.21)	2.54 (±0.54)	2.76 (±0.28)	2.33 (±0.59)	2.17 (±0.36)	2.01 (±0.69)	1.39 (±0.41)	2.53 (±0.47)	2.6 (±0.31)
Model 8	2.23 (±0.6)	2.12 (±0.58)	1.74 (±0.54)	2.53 (±0.53)	2.46 (±0.53)	2.63 (±0.47)	2.24 (±0.59)	2.13 (±0.49)	1.91 (±0.68)	1.75 (±0.51)	2.24 (±0.5)	2.51 (±0.38)
Model 9	2.34 (±0.62)	2.31 (±0.61)	1.7 (±0.67)	2.72 (±0.24)	2.46 (±0.54)	2.7 (±0.34)	2.51 (±0.41)	2.17 (±0.37)	2.33 (±0.54)	1.34 (±0.42)	2.58 (±0.4)	2.61 (±0.33)
Model 10	2.36 (±0.76)	1.95 (±0.69)	1.98 (±0.7)	**2.75 (±0.19)**	2.74 (±0.35)	2.64 (±0.38)	2.47 (±0.49)	1.98 (±0.26)	2.56 (±0.36)	1.35 (±0.3)	2.48 (±0.5)	**2.82 (±0.24)**

Table 5.10: Fitness values for base learner models of the 3^{rd} framework averaged over the 100 repetitions using Eq. 3.8 and data division Case B.

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Fold 11	Fold 12
Model 1	1.82 (±0.69)	1.82 (±0.52)	1.51 (±0.56)	2.51 (±0.37)	2.53 (±0.39)	2.63 (±0.26)	2.33 (±0.45)	1.61 (±0.31)	1.9 (±0.51)	1.26 (±0.25)	2.2 (±0.39)	2.66 (±0.2)
Model 2	2.13 (±0.66)	1.93 (±0.59)	1.5 (±0.49)	2.56 (±0.3)	2.54 (±0.38)	2.53 (±0.44)	2.34 (±0.43)	1.65 (±0.35)	1.99 (±0.48)	1.28 (±0.24)	2.32 (±0.45)	2.58 (±0.34)
Model 3	2.44 (±0.48)	1.89 (±0.57)	1.42 (±0.55)	2.48 (±0.34)	2.46 (±0.37)	2.54 (±0.34)	2.46 (±0.35)	1.81 (±0.38)	1.82 (±0.61)	1.39 (±0.31)	2.35 (±0.47)	2.55 (±0.3)
Model 4	1.96 (±0.63)	1.79 (±0.5)	1.44 (±0.54)	2.49 (±0.39)	2.45 (±0.44)	2.56 (±0.37)	2.41 (±0.4)	1.7 (±0.33)	1.88 (±0.49)	1.3 (±0.27)	2.11 (±0.49)	2.55 (±0.28)
Model 5	2.51 (±0.7)	1.46 (±0.47)	2.03 (±0.5)	2.48 (±0.35)	2.7 (±0.24)	2.43 (±0.42)	2.25 (±0.44)	1.59 (±0.29)	2.07 (±0.36)	1.26 (±0.21)	2.19 (±0.45)	2.72 (±0.22)
Model 6	1.84 (±0.71)	1.49 (±0.47)	1.86 (±0.46)	2.1 (±0.54)	2.37 (±0.46)	2.45 (±0.27)	2.2 (±0.45)	1.48 (±0.28)	1.75 (±0.49)	1.27 (±0.26)	1.79 (±0.43)	2.43 (±0.37)
Model 7	2.2 (±0.69)	1.86 (±0.54)	1.52 (±0.5)	2.57 (±0.29)	2.49 (±0.45)	2.64 (±0.29)	2.23 (±0.51)	1.76 (±0.33)	1.8 (±0.56)	1.27 (±0.25)	2.33 (±0.42)	2.59 (±0.28)
Model 8	1.86 (±0.56)	1.66 (±0.51)	1.33 (±0.46)	2.24 (±0.5)	2.41 (±0.45)	2.44 (±0.4)	2.06 (±0.5)	1.69 (±0.42)	1.7 (±0.54)	1.43 (±0.38)	2.05 (±0.44)	2.48 (±0.34)
Model 9	2.19 (±0.64)	2.03 (±0.52)	1.48 (±0.53)	2.51 (±0.29)	2.45 (±0.45)	2.55 (±0.38)	2.38 (±0.42)	1.75 (±0.36)	2.06 (±0.48)	1.3 (±0.25)	2.37 (±0.38)	2.6 (±0.3)
Model 10	2.26 (±0.76)	1.7 (±0.54)	1.76 (±0.58)	2.52 (±0.31)	2.68 (±0.31)	2.41 (±0.42)	2.24 (±0.46)	1.63 (±0.22)	2.08 (±0.4)	1.34 (±0.22)	2.21 (±0.4)	**2.75 (±0.21)**

division Case B and Case C can be seen in Figure 5.26(a) and (b), respectively. The bright green line represents the ideal HI, the dotted blue lines are the training SSPs, and the black line is the test SSP. At first glance, it is observed that not every fold is able to create a HI close to the ideal. For instance, in Fold 3, not even the training units are able to fit the ideal HI, although the different units display a similar trend. On the other hand, for Fold 10, the training SSPs display a behavior close to the ideal one; however, the test unit fails to reproduce this trend. Contrary to Figure 5.26(a) and data division Case B, the test unit deviates from the training units quite significantly and in more Folds in Figure 5.26(b)

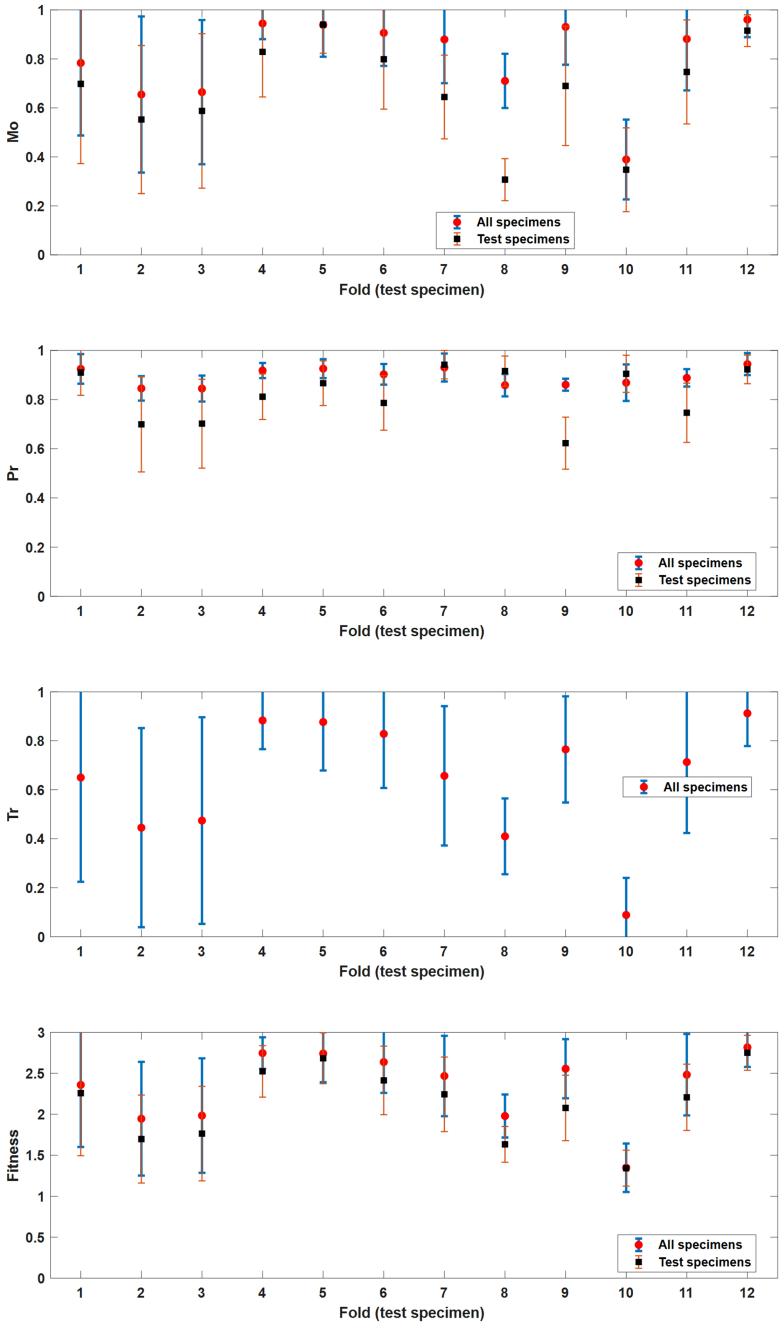


Figure 5.24: The prognostic metrics distribution based on all and test units for the base learner Model 10 of the 3^{rd} framework, considering Case B for the dataset division (the test SSP itself as validation).

Table 5.11: Fitness values for base learner models of the 3^{rd} framework averaged over the 100 repetitions using Eq. 3.4 and data division Case C.

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Fold 11	Fold 12
Model 1	1.8 (±0.55)	2 (±0.59)	2.08 (±0.61)	2.51 (±0.38)	2.18 (±0.62)	2.03 (±0.65)	1.67 (±0.5)	1.93 (±0.54)	2.02 (±0.49)	1.96 (±0.36)	2.06 (±0.51)	2.43 (±0.46)
Model 2	1.72 (±0.56)	2 (±0.6)	2.07 (±0.55)	2 (±0.67)	2.21 (±0.55)	2.09 (±0.6)	1.85 (±0.57)	2.21 (±0.49)	2.2 (±0.41)	1.94 (±0.45)	1.92 (±0.57)	2.38 (±0.56)
Model 3	2.07 (±0.49)	2.03 (±0.54)	2.12 (±0.54)	2.19 (±0.55)	2.19 (±0.59)	2.08 (±0.65)	1.95 (±0.59)	1.65 (±0.52)	2.1 (±0.5)	1.99 (±0.45)	2.09 (±0.53)	2.18 (±0.61)
Model 4	1.97 (±0.46)	1.96 (±0.64)	1.86 (±0.55)	2.19 (±0.55)	1.98 (±0.67)	2.21 (±0.69)	1.85 (±0.61)	1.99 (±0.55)	2.03 (±0.45)	1.83 (±0.47)	2.02 (±0.53)	2.27 (±0.55)
Model 5	1.6 (±0.52)	2.17 (±0.5)	2.01 (±0.6)	2.17 (±0.59)	2.2 (±0.61)	2.04 (±0.55)	1.56 (±0.4)	2.02 (±0.37)	1.95 (±0.58)	1.8 (±0.42)	2 (±0.54)	2.07 (±0.55)
Model 6	1.71 (±0.4)	1.88 (±0.53)	1.84 (±0.51)	2.02 (±0.53)	2.11 (±0.52)	1.92 (±0.48)	1.73 (±0.4)	1.79 (±0.41)	1.87 (±0.4)	1.86 (±0.41)	1.83 (±0.42)	1.92 (±0.57)
Model 7	2.09 (±0.51)	2.15 (±0.58)	2.05 (±0.52)	2.26 (±0.57)	2.35 (±0.53)	2.3 (±0.57)	1.75 (±0.53)	2.2 (±0.53)	1.97 (±0.54)	2.1 (±0.39)	2.08 (±0.53)	2.33 (±0.55)
Model 8	1.8 (±0.45)	2.02 (±0.55)	1.7 (±0.45)	2.21 (±0.51)	2.01 (±0.59)	2.17 (±0.56)	1.75 (±0.48)	2.11 (±0.49)	1.97 (±0.47)	1.82 (±0.39)	1.94 (±0.51)	2.17 (±0.58)
Model 9	1.95 (±0.55)	2.22 (±0.51)	2.12 (±0.56)	2.27 (±0.56)	2.36 (±0.5)	2.21 (±0.67)	1.91 (±0.53)	1.86 (±0.62)	2.03 (±0.51)	1.93 (±0.5)	2.15 (±0.56)	**2.51 (±0.43)**
Model 10	1.97 (±0.56)	2.18 (±0.51)	2.05 (±0.5)	1.93 (±0.55)	2.1 (±0.65)	2.1 (±0.58)	1.78 (±0.5)	1.92 (±0.47)	1.92 (±0.45)	1.8 (±0.42)	1.96 (±0.48)	2.25 (±0.57)

Table 5.12: Fitness values for base learner models of the 3^{rd} framework averaged over the 100 repetitions using Eq. 3.8 and data division Case C.

	Fold 1	Fold 2	Fold 3	Fold 4	Fold 5	Fold 6	Fold 7	Fold 8	Fold 9	Fold 10	Fold 11	Fold 12
Model 1	1.59 (±0.54)	1.62 (±0.52)	1.68 (±0.55)	2.15 (±0.55)	2 (±0.59)	1.91 (±0.55)	1.5 (±0.41)	1.49 (±0.52)	1.47 (±0.53)	1.33 (±0.26)	1.75 (±0.43)	2.38 (±0.46)
Model 2	1.45 (±0.56)	1.68 (±0.53)	1.57 (±0.6)	1.81 (±0.64)	2.03 (±0.57)	1.94 (±0.6)	1.58 (±0.51)	1.75 (±0.5)	1.58 (±0.65)	1.43 (±0.34)	1.7 (±0.5)	2.32 (±0.55)
Model 3	1.77 (±0.63)	1.63 (±0.51)	1.64 (±0.57)	1.85 (±0.6)	2.08 (±0.54)	1.94 (±0.61)	1.79 (±0.58)	1.37 (±0.48)	1.62 (±0.6)	1.45 (±0.32)	1.84 (±0.49)	2.16 (±0.6)
Model 4	1.58 (±0.56)	1.6 (±0.57)	1.41 (±0.52)	1.89 (±0.58)	1.89 (±0.6)	2.12 (±0.62)	1.64 (±0.59)	1.59 (±0.5)	1.5 (±0.57)	1.37 (±0.28)	1.69 (±0.46)	2.2 (±0.55)
Model 5	1.38 (±0.5)	1.77 (±0.41)	1.63 (±0.54)	1.89 (±0.62)	2.05 (±0.6)	1.78 (±0.53)	1.3 (±0.28)	1.33 (±0.53)	1.52 (±0.53)	1.3 (±0.29)	1.73 (±0.46)	1.99 (±0.55)
Model 6	1.38 (±0.42)	1.47 (±0.48)	1.42 (±0.54)	1.62 (±0.55)	2.05 (±0.51)	1.67 (±0.49)	1.5 (±0.37)	1.31 (±0.39)	1.31 (±0.48)	1.36 (±0.37)	1.49 (±0.35)	1.8 (±0.55)
Model 7	1.76 (±0.59)	1.8 (±0.55)	1.52 (±0.58)	1.95 (±0.64)	2.12 (±0.55)	2.07 (±0.53)	1.49 (±0.47)	1.81 (±0.52)	1.48 (±0.57)	1.53 (±0.38)	1.79 (±0.52)	2.25 (±0.52)
Model 8	1.29 (±0.45)	1.59 (±0.5)	1.15 (±0.44)	1.76 (±0.54)	1.85 (±0.52)	1.94 (±0.56)	1.48 (±0.44)	1.62 (±0.48)	1.46 (±0.54)	1.23 (±0.33)	1.63 (±0.5)	2.03 (±0.64)
Model 9	1.67 (±0.59)	1.77 (±0.55)	1.62 (±0.65)	1.98 (±0.59)	2.12 (±0.54)	2.06 (±0.63)	1.64 (±0.51)	1.5 (±0.58)	1.49 (±0.6)	1.46 (±0.3)	1.84 (±0.54)	**2.44 (±0.4)**
Model 10	1.68 (±0.65)	1.75 (±0.46)	1.58 (±0.54)	1.59 (±0.59)	1.94 (±0.65)	1.89 (±0.51)	1.49 (±0.42)	1.37 (±0.46)	1.29 (±0.51)	1.25 (±0.25)	1.63 (±0.41)	2.19 (±0.57)

(data division Case C). This is also observed by the drop in average Fitness values in Table 5.12. This comments that the base learners are struggling to generalize the good results of training with unseen data.

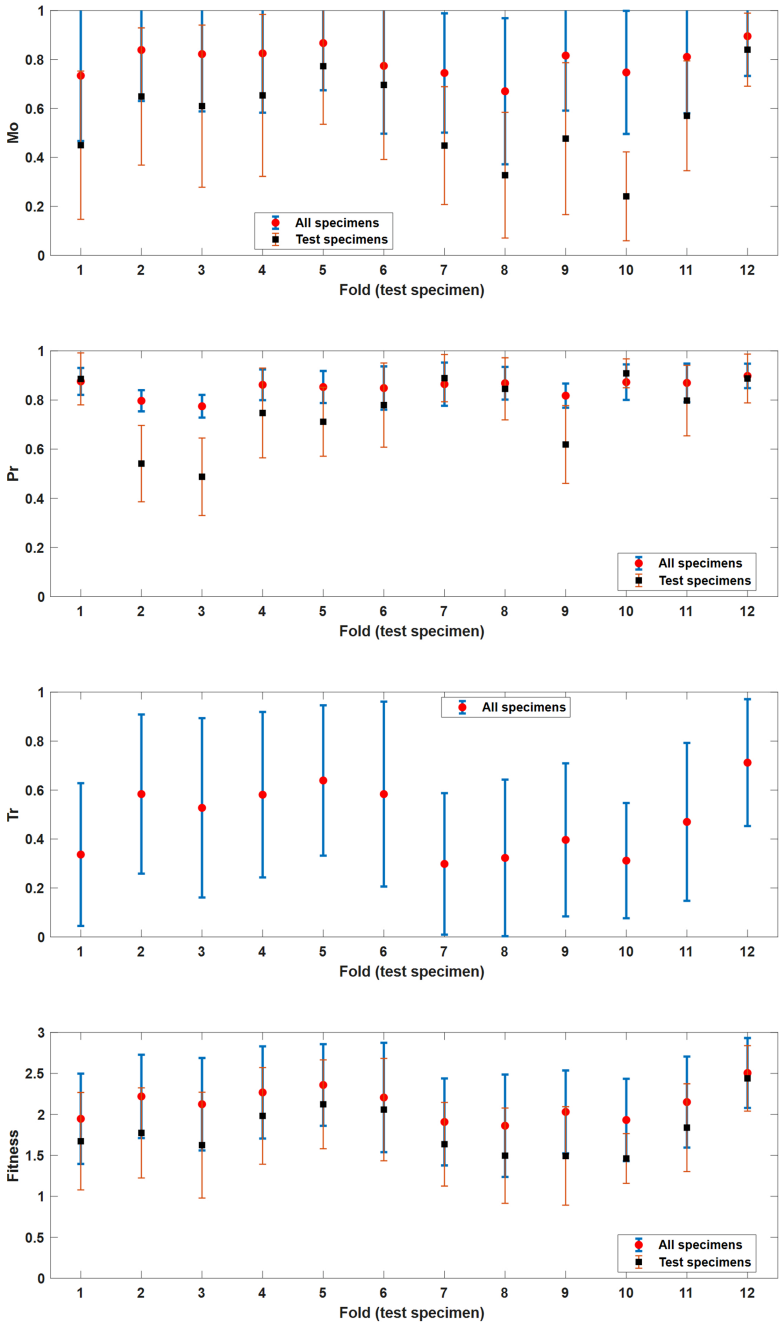
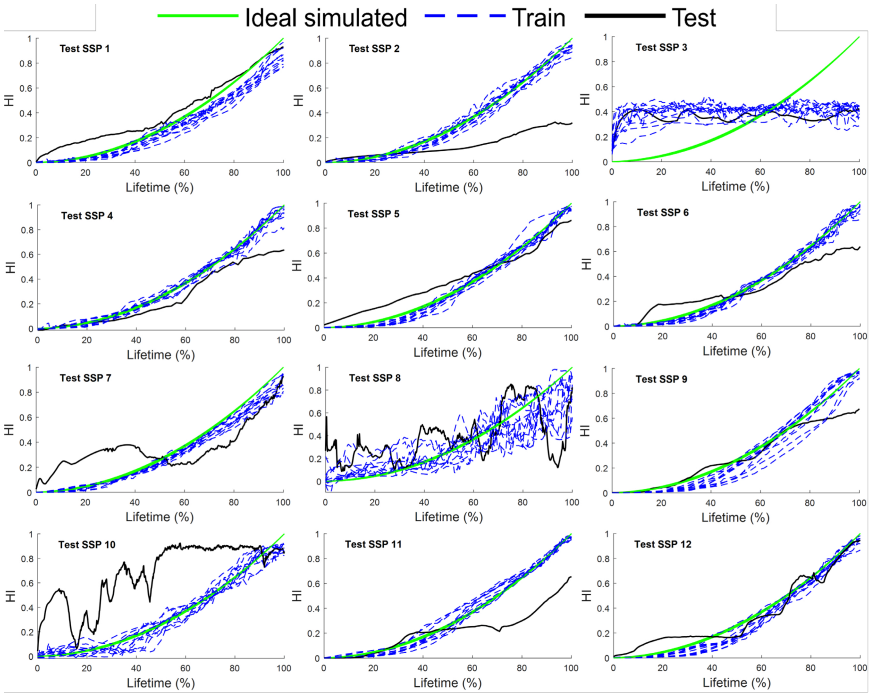
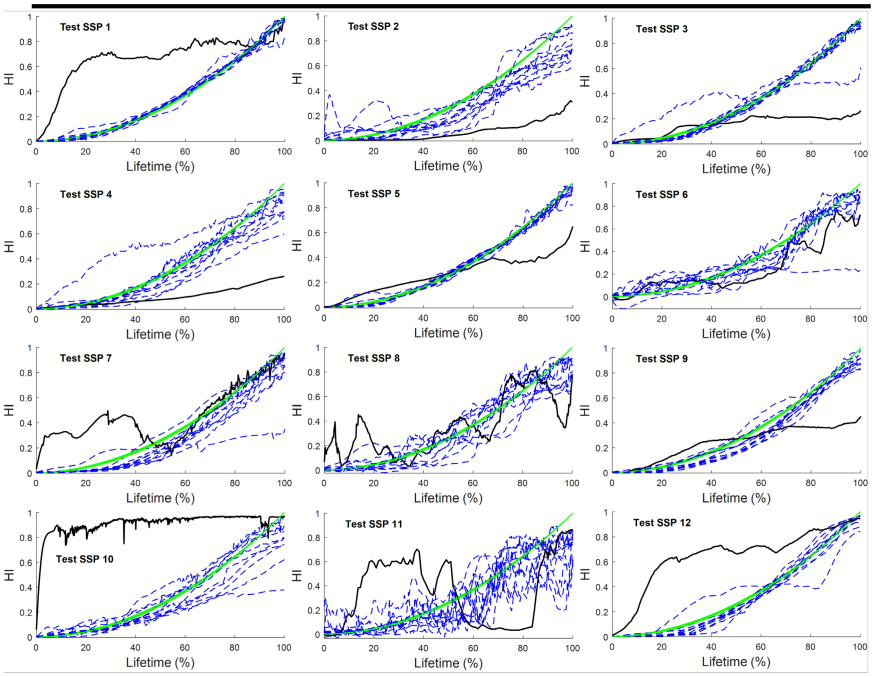


Figure 5.25: The prognostic metrics distribution based on all and test units for the base learner Model 10 of the 3^{rd} framework, considering Case C for the dataset division (another SSP other than the test SSP as validation).



(a)



(b)

Figure 5.26: (a) The HIs constructed by the base Model 10 of the 3^{rd} framework, considering Case B for the dataset division (the test SSP itself as validation), with Fitness 2.42 (±0.52) based on Eq. 3.8; (b) The HIs constructed by the base Model 9 of the 3^{rd} framework, considering Case C for the dataset division (another SSP other than the test SSP as validation), with Fitness 2.21 (±0.39) based on Eq. 3.8.

Ensemble learner models

In Table 5.13, the ensemble models' Fitness averaged over the 12 Folds is presented for the different ensemble methodologies, where the base learner is trained with Case B (test unit as validation). When considering the entire set (training and test) it is observed that Fitness is mostly over 2.6. What is surprising is the performance of WAE-MSE and WAE-RMSE, which are underperforming even compared to SAE. The best Fitness result is obtained using the 16^{th} EL model, Net(12), which is one of the more complex networks, where the Fitness value is 2.84 (±0.13) and is obtained for base learner Model 9, while base learner Model 3 also achieves a similar value of 2.84 (±0.15).

As it was previously mentioned, it is important to calculate the prognostic metrics for the test unit and evaluate the performance of the models only on unknown data. Table 5.14 demonstrates the Fitness values calculated using Eq. 3.8. The overall values have decreased, which is expected since the test unit does not fit the ideal behavior perfectly and diverges from the average behavior of the training units. The best Fitness is again obtained for the Net(12) ensemble upon Model 3, with a value of 2.74 (±0.33). The results of Tables 5.13 and 5.14 are also visually summarized in Figure 5.27 for Model 3. It is evident that the overall Fitness has only slightly reduced when only considering the test units, which demonstrates the effectiveness of the methodology.

Table 5.13: Fitness values for ensemble learner models of the 3^{rd} framework averaged over the 12 Folds using Eq. 3.4 and data division Case B.

		WAE			DL-based: Net											
	SAE	MSE	RMSE	Fitness	(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)
Model 1	2.69 (±0.26)	2.37 (±0.38)	2.61 (±0.29)	2.7 (±0.24)	2.39 (±0.39)	2.75 (±0.19)	2.68 (±0.13)	2.72 (±0.24)	2.72 (±0.31)	2.74 (±0.22)	2.71 (±0.25)	2.63 (±0.18)	2.66 (±0.17)	2.7 (±0.18)	2.76 (±0.22)	2.78 (±0.24)
Model 2	2.75 (±0.19)	2.52 (±0.27)	2.68 (±0.21)	2.75 (±0.18)	2.49 (±0.34)	2.76 (±0.18)	2.74 (±0.12)	2.72 (±0.14)	2.79 (±0.17)	2.75 (±0.2)	2.73 (±0.19)	2.57 (±0.23)	2.66 (±0.13)	2.72 (±0.13)	2.78 (±0.19)	2.81 (±0.18)
Model 3	2.75 (±0.14)	2.51 (±0.34)	2.67 (±0.2)	2.76 (±0.13)	2.53 (±0.39)	2.78 (±0.12)	2.63 (±0.19)	2.68 (±0.17)	2.82 (±0.11)	2.79 (±0.13)	2.67 (±0.19)	2.65 (±0.13)	2.7 (±0.09)	2.75 (±0.11)	2.81 (±0.14)	**2.84 (±0.15)**
Model 4	2.73 (±0.17)	2.44 (±0.27)	2.64 (±0.22)	2.73 (±0.16)	2.69 (±0.21)	2.79 (±0.12)	2.7 (±0.15)	2.72 (±0.12)	2.78 (±0.13)	2.77 (±0.14)	2.72 (±0.14)	2.64 (±0.11)	2.67 (±0.11)	2.72 (±0.11)	2.71 (±0.32)	2.82 (±0.14)
Model 5	2.74 (±0.22)	2.57 (±0.24)	2.7 (±0.21)	2.74 (±0.22)	2.57 (±0.2)	2.76 (±0.17)	2.63 (±0.18)	2.71 (±0.22)	2.8 (±0.14)	2.77 (±0.17)	2.76 (±0.17)	2.63 (±0.16)	2.64 (±0.15)	2.74 (±0.09)	2.78 (±0.2)	2.8 (±0.19)
Model 6	2.62 (±0.27)	2.32 (±0.31)	2.47 (±0.27)	2.63 (±0.26)	2.41 (±0.38)	2.73 (±0.24)	2.46 (±0.34)	2.66 (±0.28)	2.74 (±0.21)	2.74 (±0.23)	2.7 (±0.21)	2.54 (±0.25)	2.63 (±0.18)	2.7 (±0.14)	2.77 (±0.13)	2.77 (±0.17)
Model 7	2.75 (±0.2)	2.41 (±0.38)	2.64 (±0.26)	2.76 (±0.17)	2.66 (±0.15)	2.78 (±0.13)	2.58 (±0.22)	2.66 (±0.37)	2.77 (±0.14)	2.81 (±0.14)	2.74 (±0.18)	2.62 (±0.12)	2.68 (±0.09)	2.73 (±0.11)	2.81 (±0.14)	2.84 (±0.14)
Model 8	2.74 (±0.13)	2.15 (±0.28)	2.59 (±0.15)	2.74 (±0.13)	2.63 (±0.23)	2.75 (±0.14)	2.68 (±0.15)	2.72 (±0.14)	2.77 (±0.13)	2.75 (±0.15)	2.7 (±0.21)	2.66 (±0.16)	2.65 (±0.12)	2.72 (±0.09)	2.77 (±0.15)	2.77 (±0.17)
Model 9	2.76 (±0.15)	2.6 (±0.19)	2.71 (±0.15)	2.76 (±0.15)	2.47 (±0.38)	2.75 (±0.17)	2.62 (±0.23)	2.62 (±0.34)	2.79 (±0.13)	2.75 (±0.16)	2.71 (±0.2)	2.62 (±0.18)	2.66 (±0.15)	2.72 (±0.13)	2.76 (±0.19)	**2.84 (±0.13)**
Model 10	2.75 (±0.24)	2.53 (±0.24)	2.7 (±0.23)	2.75 (±0.23)	2.64 (±0.26)	2.77 (±0.15)	2.61 (±0.21)	2.63 (±0.36)	2.77 (±0.15)	2.76 (±0.14)	2.7 (±0.3)	2.61 (±0.17)	2.67 (±0.13)	2.72 (±0.1)	2.8 (±0.15)	2.79 (±0.21)

Using a random SSP as the validation unit (data division case C) slightly affects the Fitness values of the HIs. Tables 5.15 and 5.16 show the average Fitness values (and standard deviation) averaged over the 12 Folds, calculated for the entire set (using Eq. 3.4) and only for the test unit (using Eq. 3.8) respectively. The highest Fitness value is obtained by Net(12) upon Model 9 in both cases with values 2.74 (±0.19) and 2.59 (±0.24), respectively. As expected, the Fitness value is lower than in the respective cases where the test unit is used in the training, since in this case the test unit is unknown. This case, however, is more representative of the potential real-world application of the methodology. In Figure 5.28, the results of Tables 5.15 and 5.16 for base learner Model 9 are visually summarized and

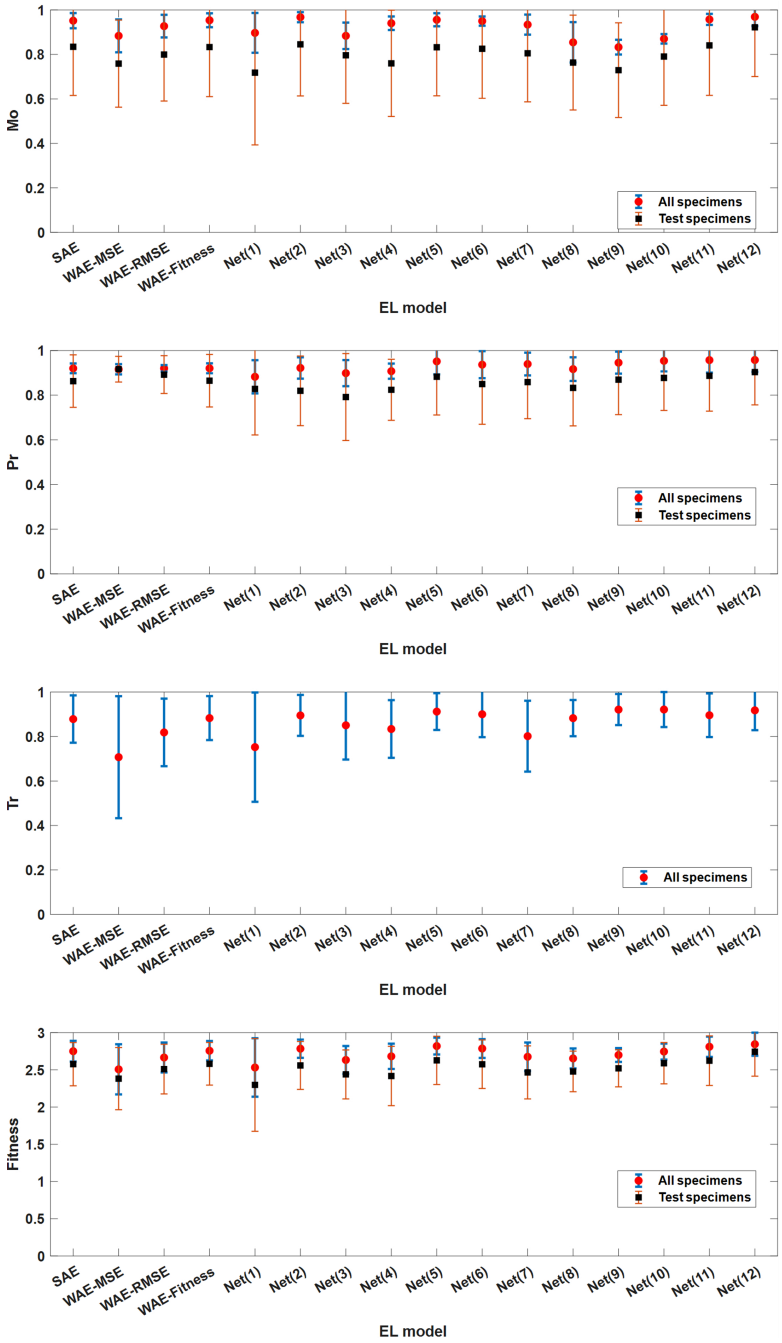


Figure 5.27: The prognostic metrics distribution based on all and test units for ensemble learner models upon the base learner Model 3 of the 3^{rd} framework, considering Case B for the dataset division (the test SSP itself as validation).

Table 5.14: Fitness values for ensemble learner models of the 3^{rd} framework averaged over the 12 Folds using Eq. 3.8 and data division Case B.

		WAE			DL-based: Net											
	SAE	**MSE**	**RMSE**	**Fitness**	**(1)**	**(2)**	**(3)**	**(4)**	**(5)**	**(6)**	**(7)**	**(8)**	**(9)**	**(10)**	**(11)**	**(12)**
Model 1	2.44 (±0.39)	2.21 (±0.43)	2.39 (±0.41)	2.45 (±0.38)	2.09 (±0.66)	2.44 (±0.41)	2.42 (±0.29)	2.44 (±0.44)	2.48 (±0.51)	2.47 (±0.44)	2.41 (±0.52)	2.33 (±0.37)	2.43 (±0.3)	2.5 (±0.37)	2.5 (±0.45)	2.6 (±0.44)
Model 2	2.53 (±0.33)	2.39 (±0.34)	2.49 (±0.34)	2.53 (±0.33)	2.2 (±0.53)	2.48 (±0.4)	2.51 (±0.31)	2.47 (±0.26)	2.55 (±0.4)	2.51 (±0.4)	2.48 (±0.45)	2.3 (±0.41)	2.44 (±0.31)	2.56 (±0.29)	2.54 (±0.39)	2.67 (±0.37)
Model 3	2.58 (±0.29)	2.38 (±0.42)	2.51 (±0.33)	2.58 (±0.29)	2.3 (±0.62)	2.56 (±0.32)	2.44 (±0.33)	2.42 (±0.4)	2.63 (±0.32)	2.57 (±0.33)	2.47 (±0.36)	2.48 (±0.27)	2.52 (±0.25)	2.59 (±0.28)	2.62 (±0.33)	**2.74 (±0.33)**
Model 4	2.5 (±0.32)	2.29 (±0.35)	2.44 (±0.34)	2.49 (±0.31)	2.44 (±0.4)	2.49 (±0.32)	2.35 (±0.39)	2.44 (±0.35)	2.5 (±0.36)	2.5 (±0.36)	2.45 (±0.36)	2.4 (±0.28)	2.44 (±0.29)	2.53 (±0.29)	2.4 (±0.63)	2.71 (±0.21)
Model 5	2.5 (±0.36)	2.39 (±0.33)	2.48 (±0.36)	2.5 (±0.37)	2.24 (±0.47)	2.44 (±0.43)	2.27 (±0.41)	2.41 (±0.47)	2.52 (±0.4)	2.48 (±0.42)	2.46 (±0.43)	2.37 (±0.34)	2.39 (±0.36)	2.55 (±0.23)	2.54 (±0.44)	2.61 (±0.35)
Model 6	2.39 (±0.38)	2.15 (±0.35)	2.27 (±0.34)	2.4 (±0.37)	2.16 (±0.41)	2.41 (±0.43)	2.05 (±0.65)	2.38 (±0.41)	2.45 (±0.46)	2.47 (±0.45)	2.4 (±0.4)	2.25 (±0.32)	2.39 (±0.36)	2.52 (±0.23)	2.55 (±0.25)	2.66 (±0.26)
Model 7	2.54 (±0.37)	2.29 (±0.43)	2.47 (±0.41)	2.55 (±0.36)	2.42 (±0.31)	2.51 (±0.35)	2.33 (±0.39)	2.45 (±0.47)	2.55 (±0.4)	2.61 (±0.33)	2.51 (±0.41)	2.36 (±0.27)	2.5 (±0.21)	2.55 (±0.29)	2.59 (±0.32)	2.72 (±0.29)
Model 8	2.46 (±0.27)	2.1 (±0.28)	2.4 (±0.25)	2.46 (±0.27)	2.3 (±0.39)	2.39 (±0.35)	2.32 (±0.33)	2.48 (±0.28)	2.5 (±0.32)	2.46 (±0.33)	2.44 (±0.43)	2.36 (±0.28)	2.42 (±0.25)	2.52 (±0.2)	2.48 (±0.34)	2.57 (±0.33)
Model 9	2.55 (±0.3)	2.46 (±0.25)	2.52 (±0.27)	2.54 (±0.32)	2.19 (±0.38)	2.47 (±0.42)	2.34 (±0.45)	2.37 (±0.49)	2.56 (±0.37)	2.51 (±0.37)	2.46 (±0.43)	2.37 (±0.32)	2.45 (±0.36)	2.54 (±0.31)	2.53 (±0.43)	2.69 (±0.3)
Model 10	2.51 (±0.36)	2.37 (±0.34)	2.48 (±0.35)	2.51 (±0.36)	2.34 (±0.46)	2.47 (±0.37)	2.3 (±0.42)	2.34 (±0.52)	2.55 (±0.34)	2.51 (±0.34)	2.46 (±0.5)	2.33 (±0.37)	2.44 (±0.3)	2.52 (±0.27)	2.57 (±0.36)	2.61 (±0.38)

compared. The values of the test-only-based Fitness are slightly lower than those for the entire set, and especially for Net(12) ensemble, which displays promising results.

Table 5.15: Fitness values for ensemble learner models of the 3^{rd} framework averaged over the 12 Folds using Eq. 3.4 and data division Case C.

		WAE			DL-based: Net											
	SAE	**MSE**	**RMSE**	**Fitness**	**(1)**	**(2)**	**(3)**	**(4)**	**(5)**	**(6)**	**(7)**	**(8)**	**(9)**	**(10)**	**(11)**	**(12)**
Model 1	2.62 (±0.2)	2.42 (±0.22)	2.54 (±0.2)	2.62 (±0.19)	2.43 (±0.35)	2.61 (±0.19)	2.52 (±0.28)	2.56 (±0.19)	2.64 (±0.2)	2.59 (±0.19)	2.52 (±0.28)	2.41 (±0.34)	2.5 (±0.21)	2.56 (±0.2)	2.64 (±0.19)	2.63 (±0.2)
Model 2	2.63 (±0.17)	2.48 (±0.23)	2.58 (±0.19)	2.63 (±0.17)	2.35 (±0.28)	2.65 (±0.16)	2.5 (±0.15)	2.55 (±0.24)	2.63 (±0.19)	2.58 (±0.2)	2.61 (±0.18)	2.48 (±0.19)	2.52 (±0.21)	2.59 (±0.17)	2.61 (±0.19)	2.67 (±0.27)
Model 3	2.67 (±0.12)	2.53 (±0.11)	2.62 (±0.12)	2.67 (±0.12)	2.25 (±0.49)	2.69 (±0.11)	2.5 (±0.34)	2.6 (±0.19)	2.69 (±0.14)	2.62 (±0.17)	2.6 (±0.18)	2.58 (±0.13)	2.58 (±0.13)	2.62 (±0.15)	2.67 (±0.18)	2.7 (±0.17)
Model 4	2.64 (±0.17)	2.41 (±0.24)	2.55 (±0.19)	2.64 (±0.17)	2.24 (±0.42)	2.64 (±0.16)	2.52 (±0.23)	2.54 (±0.21)	2.62 (±0.21)	2.6 (±0.22)	2.53 (±0.35)	2.42 (±0.33)	2.54 (±0.16)	2.59 (±0.16)	2.59 (±0.23)	2.64 (±0.31)
Model 5	2.55 (±0.3)	2.47 (±0.18)	2.5 (±0.24)	2.54 (±0.3)	2.37 (±0.36)	2.52 (±0.31)	2.34 (±0.29)	2.44 (±0.34)	2.55 (±0.3)	2.51 (±0.29)	2.52 (±0.27)	2.44 (±0.3)	2.47 (±0.32)	2.52 (±0.29)	2.58 (±0.28)	2.59 (±0.26)
Model 6	2.43 (±0.24)	2.18 (±0.24)	2.28 (±0.25)	2.44 (±0.24)	1.94 (±0.41)	2.53 (±0.23)	2.31 (±0.28)	2.36 (±0.3)	2.54 (±0.27)	2.51 (±0.23)	2.36 (±0.39)	2.32 (±0.29)	2.43 (±0.29)	2.49 (±0.25)	2.6 (±0.23)	2.61 (±0.21)
Model 7	2.65 (±0.15)	2.42 (±0.23)	2.57 (±0.18)	2.65 (±0.16)	2.42 (±0.38)	2.66 (±0.16)	2.37 (±0.25)	2.53 (±0.26)	2.64 (±0.21)	2.63 (±0.17)	2.56 (±0.19)	2.49 (±0.28)	2.54 (±0.17)	2.6 (±0.16)	2.67 (±0.18)	2.69 (±0.16)
Model 8	2.56 (±0.22)	2.17 (±0.27)	2.44 (±0.25)	2.56 (±0.21)	2.4 (±0.29)	2.57 (±0.25)	2.37 (±0.37)	2.45 (±0.24)	2.57 (±0.24)	2.56 (±0.23)	2.45 (±0.34)	2.52 (±0.23)	2.47 (±0.24)	2.56 (±0.2)	2.59 (±0.22)	2.61 (±0.34)
Model 9	2.67 (±0.15)	2.47 (±0.24)	2.6 (±0.17)	2.67 (±0.15)	2.55 (±0.22)	2.67 (±0.16)	2.53 (±0.23)	2.6 (±0.23)	2.64 (±0.19)	2.6 (±0.21)	2.61 ±0.22)	2.43 (±0.35)	2.53 (±0.19)	2.62 (±0.16)	2.62 (±0.25)	**2.74 (±0.19)**
Model 10	2.52 (±0.26)	2.41 (±0.23)	2.45 (±0.25)	2.52 (±0.26)	2.38 (±0.31)	2.56 (±0.26)	2.31 (±0.37)	2.49 (±0.29)	2.58 (±0.22)	2.54 (±0.26)	2.53 (±0.27)	2.4 (±0.3)	2.46 (±0.28)	2.51 (±0.24)	2.53 (±0.27)	2.63 (±0.22)

Figure 5.29(a) displays the HIs for the different Folds using base learner Model 9 and data division case B. Each Fold denotes the respective unit as the test. The solid gray line represents the ideal HI and the dotted colored lines represent the training unit results for each ensemble model. The solid-colored lines denote the test unit. It is observed that in some Folds, like Folds 2, 3, 9, and 10, the test unit demonstrates a different behavior than the training units and diverges from both the constructed HIs of the training units and the ideal HI. In the other Folds, the test units display a similar trend to the training. When

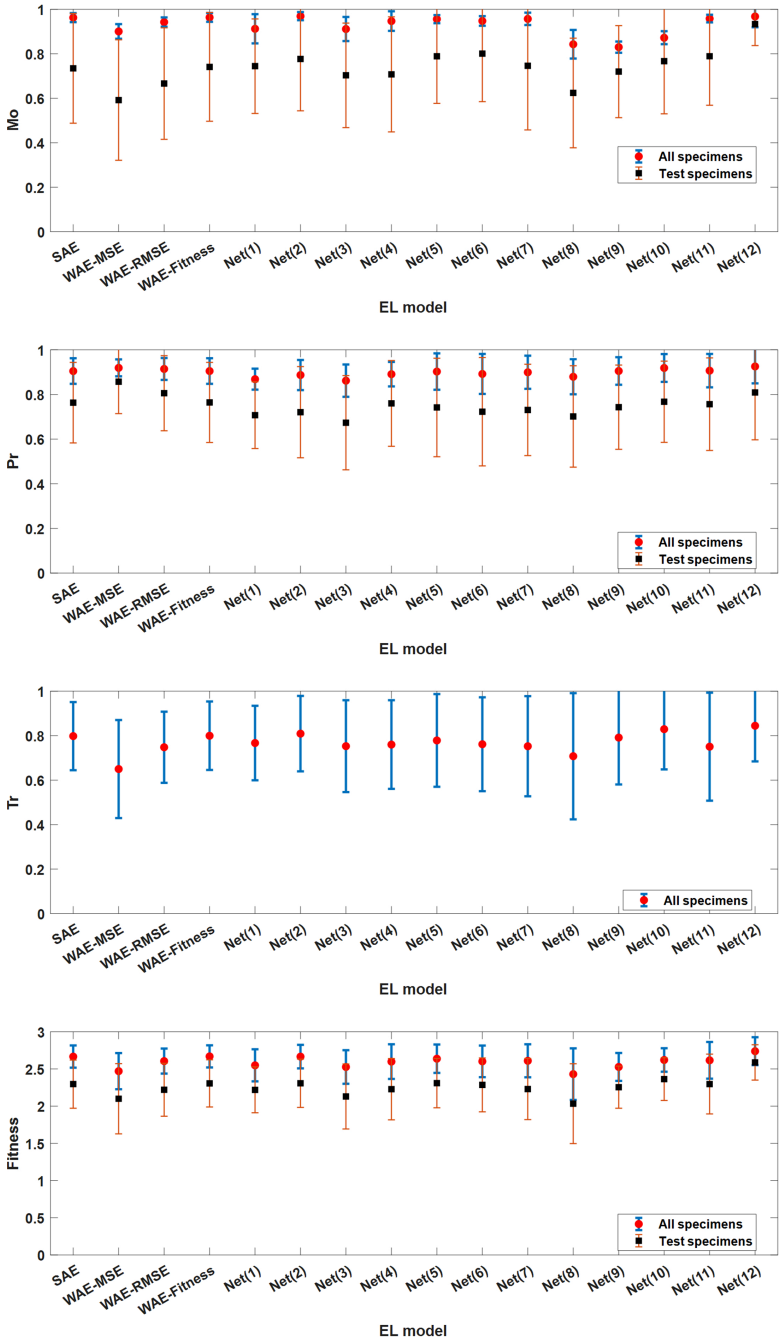


Figure 5.28: The prognostic metrics distribution based on all and test units for ensemble learner models upon the base learner Model 9 of the 3^{rd} framework, considering Case C for the dataset division (another SSP other than the test SSP as validation).

Table 5.16: Fitness values for ensemble learner models of the 3^{rd} framework averaged over the 12 Folds using Eq. 3.8 and data division Case C.

		WAE			DL-based: Net											
	SAE	**MSE**	**RMSE**	**Fitness**	**(1)**	**(2)**	**(3)**	**(4)**	**(5)**	**(6)**	**(7)**	**(8)**	**(9)**	**(10)**	**(11)**	**(12)**
Model 1	2.22 (±0.41)	2.05 (±0.44)	2.16 (±0.38)	2.23 (±0.4)	2.03 (±0.54)	2.18 (±0.41)	2.04 (±0.6)	2.13 (±0.39)	2.33 (±0.35)	2.25 (±0.35)	2.12 (±0.47)	2.02 (±0.55)	2.2 (±0.34)	2.27 (±0.34)	2.3 (±0.37)	2.43 (±0.31)
Model 2	2.23 (±0.34)	2.09 (±0.46)	2.16 (±0.4)	2.22 (±0.34)	1.86 (±0.51)	2.26 (±0.33)	2.12 (±0.34)	2.12 (±0.45)	2.23 (±0.36)	2.22 (±0.37)	2.28 (±0.35)	2.13 (±0.26)	2.22 (±0.34)	2.33 (±0.32)	2.26 (±0.37)	2.43 (±0.51)
Model 3	2.32 (±0.33)	2.18 (±0.33)	2.27 (±0.34)	2.32 (±0.33)	1.84 (±0.64)	2.38 (±0.28)	2.07 (±0.68)	2.23 (±0.41)	2.42 (±0.23)	2.32 (±0.32)	2.32 (±0.32)	2.23 (±0.3)	2.34 (±0.24)	2.41 (±0.26)	2.42 (±0.31)	2.56 (±0.29)
Model 4	2.24 (±0.39)	2.07 (±0.48)	2.18 (±0.4)	2.23 (±0.39)	1.77 (±0.55)	2.23 (±0.38)	2.02 (±0.5)	2.17 (±0.47)	2.22 (±0.43)	2.2 (±0.41)	2.07 (±0.69)	2.06 (±0.4)	2.21 (±0.35)	2.29 (±0.36)	2.23 (±0.43)	2.4 (±0.48)
Model 5	2.1 (±0.5)	2.12 (±0.36)	2.09 (±0.42)	2.09 (±0.51)	1.86 (±0.67)	2 (±0.54)	1.67 (±0.54)	1.93 (±0.56)	2.06 (±0.52)	2.02 (±0.52)	2.04 (±0.53)	2.01 (±0.46)	2.09 (±0.52)	2.15 (±0.47)	2.13 (±0.52)	2.3 (±0.48)
Model 6	1.97 (±0.45)	1.83 (±0.42)	1.89 (±0.43)	1.98 (±0.44)	1.56 (±0.49)	2 (±0.44)	1.73 (±0.56)	1.81 (±0.57)	2.07 (±0.47)	2.05 (±0.42)	1.84 (±0.7)	1.83 (±0.5)	2.05 (±0.46)	2.08 (±0.43)	2.18 (±0.46)	2.29 (±0.43)
Model 7	2.27 (±0.3)	2.07 (±0.42)	2.19 (±0.35)	2.27 (±0.31)	1.95 (±0.56)	2.27 (±0.31)	1.92 (±0.4)	2.06 (±0.55)	2.33 (±0.35)	2.32 (±0.29)	2.23 (±0.32)	2.07 (±0.46)	2.25 (±0.31)	2.36 (±0.19)	2.36 (±0.32)	2.49 (±0.23)
Model 8	2.08 (±0.44)	1.85 (±0.46)	2.02 (±0.47)	2.08 (±0.44)	1.97 (±0.49)	2.06 (±0.49)	1.9 (±0.6)	1.97 (±0.4)	2.13 (±0.47)	2.08 (±0.47)	1.91 (±0.7)	2.07 (±0.47)	2.06 (±0.45)	2.23 (±0.38)	2.16 (±0.47)	2.28 (±0.61)
Model 9	2.3 (±0.32)	2.1 (±0.47)	2.22 (±0.36)	2.3 (±0.32)	2.22 (±0.31)	2.31 (±0.32)	2.13 (±0.44)	2.23 (±0.41)	2.31 (±0.33)	2.29 (±0.36)	2.23 (±0.41)	2.03 (±0.54)	2.25 (±0.28)	2.36 (±0.29)	2.3 (±0.4)	**2.59 (±0.24)**
Model 10	2.02 (±0.5)	2 (±0.46)	1.99 (±0.5)	2.02 (±0.49)	1.94 (±0.6)	2.08 (±0.51)	1.68 (±0.73)	1.99 (±0.54)	2.14 (±0.46)	2.09 (±0.52)	2.11 (±0.53)	1.96 (±0.49)	2.07 (±0.48)	2.11 (±0.43)	2.04 (±0.57)	2.25 (±0.49)

data division Case C is applied, the difference between training and test units increases, diverging from the ideal HI in most Folds, as displayed in Figure 5.29(b). Diverging from the ideal HI is not necessarily a negative attribute as long as training and test units display similar overall trends. The ideal HI is mostly used as a baseline for the training and not a panacea for the HIs to follow. However, there are Folds, like Folds 1 and 12 where the constructed HIs are similar between test and training as well as the ideal HI.

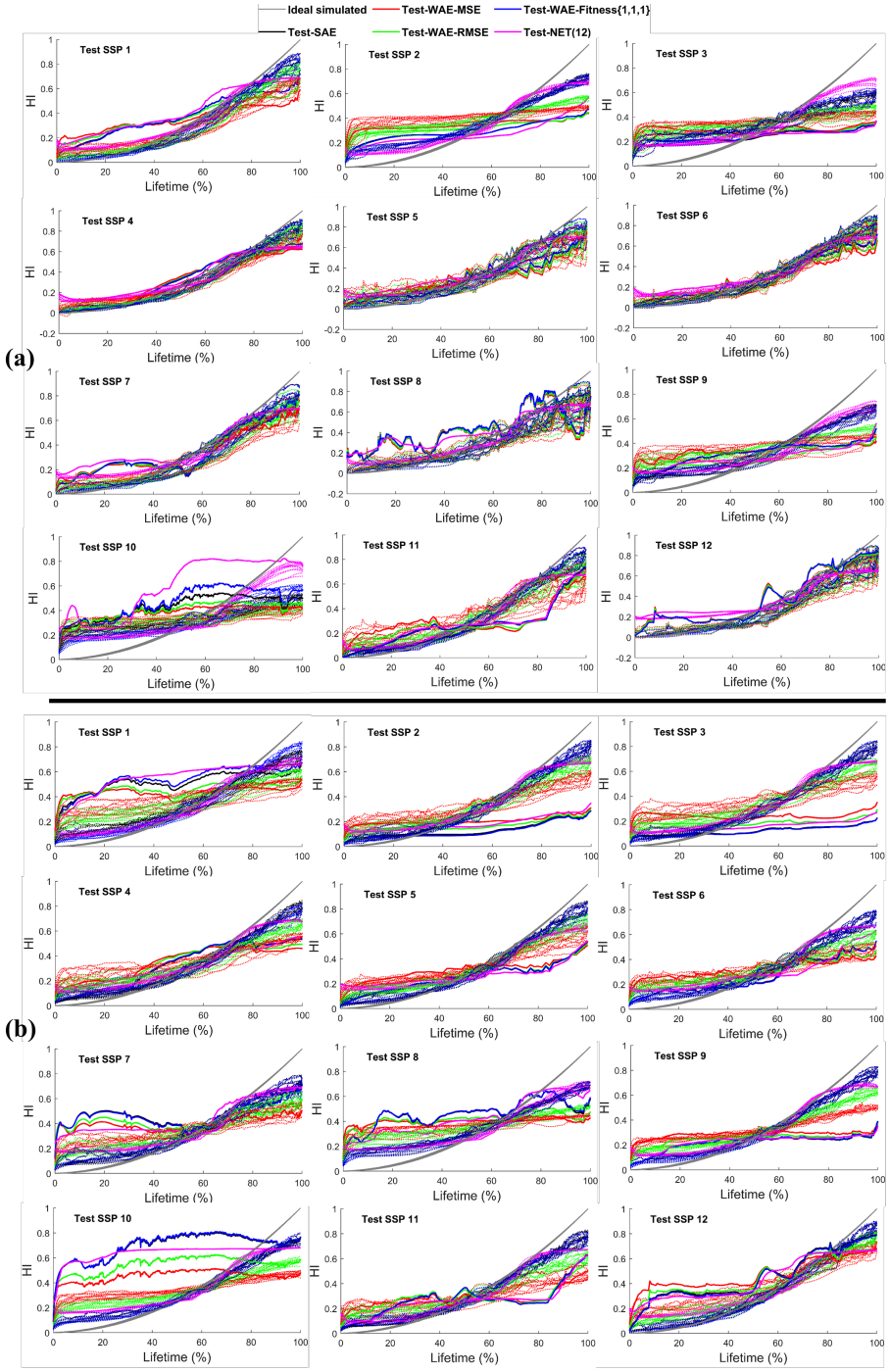


Figure 5.29: (a) The HIs constructed by the ensemble Net(12) upon the base Model 3, considering Case B for the dataset division (the test SSP itself as validation), with best average Fitness 2.74 (±0.33) based on Eq. 3.8; (b) The HIs constructed by the ensemble Net(12) upon the base Model 9, considering Case C for the dataset division (another SSP other than the test SSP as validation), with best average Fitness 2.59 (±0.24) based on Eq. 3.8. Dotted lines are related to the training units.

5.7 Discussion

It is important to note that an identified HI and its corresponding preprocessing technique are closely relevant to the engineering system/structure under monitoring [18], as well as the type of SHM system that produces signals, and as a result, they cannot be applied to other objects directly. This marks the first challenge in comparing the current work, which focuses on a new experiment, to earlier research. In several works, only Mo was reported for the utilized HIs [19–21], while only one work [22] provided all the criteria. The prognostic criteria have therefore infrequently been quantified for the reported HIs, especially for composite structures, which adds another challenge when comparing the current study with existing work. Yue et al. [22] employed the GW monitoring technique in the ReMAP project to predict the stiffness of composite panels, which is a mechanical characteristic of the structure, and then they reported the criteria for predictions and stiffness. In contrast to this chapter, they only considered five units out of twelve, and it should be highlighted that the more units used, the lower the Fitness score of HIs. One significant and critical point in the HI construction or RUL prediction frameworks, which is directly related to the Pr, is that input data must not be normalized in accordance with the mean and standard deviation of the entire dataset (training and test) [23, 24], as test data are unavailable in reality in the upcoming timeframes. A max-min normalization technique using the full dataset has similar or even more concerns [25, 26]. Several features extracted from AE data, which were considered as HI of the composite structure based on the literature (including 1/A [19, 20], energy [27], and Rise-time/Amplitude ratio (RA) [28] cumulated in the time window), will be compared with the proposed HIs. Also, the criteria for the proposed AE-based HI will be compared with the HIs extracted from GW data and mechanical properties (stiffness) of the ReMAP dataset [22]. The summarized results are presented in Table 5.17.

As can be seen, the proposed frameworks outperform the others proposed in the literature. The prognostic criteria for the windowed AE features, particularly Tr, are extremely low. The maximum Mo is for the weighted HI derived from GW data. Pr, on the other hand, is not as qualified as the proposed frameworks in the current chapter. It should also

Table 5.17: HIs' criteria (from Eqs. 3.1 – 3.4) for composite structures.

Criteria	12 composite specimens of ReMAP									5 composite specimens of ReMAP					Refs. [19, 20]*		
	Present work						According to Refs.			Ref. [22]							
	Framework 1		Framework 2		Framework 3		[19, 20]	[27]	[28]								
	AE data									L-D**	GW data				DIC data	AE data	DIC & AE
	Base model	Ensemble model	Base model	Ensemble model	Base model	Ensemble model	$\frac{1}{A}$	E	RA	stiffness	100 kHz	125 kHz	150 kHz	weighted	axial strain	$\frac{1}{A}$	Fusion
Mo	0.87 (±0.15)	0.99 (±0.01)	0.80 (±0.17)	0.97 (±0.01)	0.80 (±0.25)	0.97 (±0.05)	0.57	0.5	0.59	0.66	1	1	0.92	1	0.94	0.91	0.98
Pr	0.88 (±0.05)	0.93 (±0.03)	0.75 (±0.12)	0.84 (±0.05)	0.85 (±0.07)	0.93 (±0.08)	0.27	0.17	0.26	0.84	0.6	0.5	0.45	0.55	–	–	–
Tr	0.59 (±0.25)	0.86 (±0.14)	0.42 (±0.20)	0.88 (±0.07)	0.48 (±0.34)	0.85 (±0.16)	0	0	0	0.56	0.95	0.88	0.97	0.97	–	–	–
Fitness	2.34 (±0.45)	**2.79 (±0.14)**	1.99 (±0.31)	**2.69 (±0.07)**	2.13 (±0.57)	**2.74 (±0.19)**	0.84	0.67	0.85	2.06	2.55	2.38	2.34	2.52	–	–	–

** Different experiments to the current one*
*** The stiffness values measured from load-displacement data.*
Framework 1: Model (7) as the base model and SAE as the ensemble model
Framework 2: A.2 PCA-TIM-TDM as the base model and WAE-FitnessMSE as the ensemble model
Framework 3: Model (9) as the base model and Net(12) as the ensemble model

be highlighted that, even though both the AE and GW systems rely on acoustic and elastic waves within the structure, one is passive and the other is active. As a result, they measure different structural characteristics, resulting in a variety of data spaces. Therefore, the GW system may provide more informative data, whereas the physical model utilized in Ref. [22] may not have been able to build its best HI. As a result, Table 5.17 compares the data's informativity as well as the proposed models for constructing HIs. Although only Mo has been quantitatively reported for the HIs obtained from DIC data, AE data, and the predefined function fusing DIC and AE data in Refs. [19, 20], not only is Mo of the HIs developed in the current work slightly higher, but Tr and Pr are also superior based on qualitative comparison.

During the project leading to this thesis, a primary workflow was established for analyzing AE data, leading to the development of three frameworks in a typical evolutionary journey of PhD research. Each framework was tailored to explore distinct pathways and address specific challenges encountered during the research process.

Framework 1, while showcasing superior performance, relied on the test unit for validation in the LOOCV process, potentially limiting its applicability to real-world scenarios where test units are new and unknown. Subsequent frameworks were devised to mitigate this limitation while maintaining overall performance.

Framework 2 was developed to address the validation issue and achieved comparable performance to Framework 1, while exhibiting greater stability with a standard deviation of ±0.07. Following this, Framework 3 was introduced, demonstrating further improvements in fitness compared to Framework 2. In general, the EL step improved the criteria in terms of both Fitness scores and stability

5.8 Conclusions

Three novel frameworks have been developed in this chapter for designing HIs using AE data regarding composite structures. These frameworks integrate advanced techniques for feature extraction, deep learning, and ensemble learning, aiming to enhance prognostic capabilities. Each framework introduces innovative approaches, contributing to the field of SHM. While Framework 1 shows exceptional performance, its dependency on the test unit for validation limits its applicability. Frameworks 2 and 3 address this limitation, with Framework 3 demonstrating higher Fitness scores and Framework 2 showing higher stability. Ensemble learning enhances prognostic criteria, emphasizing practical implementation potential. The results underscore the effectiveness of data-driven and artificial approaches, providing valuable insights for advancing health monitoring applications in composite structures.

References

[1] D. Zarouchas, A. Broer, G. Galanopoulos, W. Briand, R. Benedictus, and T. Loutas, "Compression compression fatigue tests on single stiffener aerospace structures," *Dataverse, NL*, 2021.

[2] M. Moradi, A. Broer, and D. Zarouchas, "Acoustic emission dataset of single-stiffener composite panels subjected to impact and run-to-failure fatigue loading," *Mendeley*,

2022.

[3] M. Moradi, A. Broer, J. Chiachío, R. Benedictus, T. H. Loutas, and D. Zarouchas, "Intelligent health indicator construction for prognostics of composite structures utilizing a semi-supervised deep neural network and shm data," *Engineering Applications of Artificial Intelligence*, vol. 117, p. 105502, 2023.

[4] M. Ge, "Analysis of source location algorithms: Part ii. iterative methods," *Journal of Acoustic Emission*, vol. 21, no. 1, pp. 29–51, 2003.

[5] A. Broer, G. Galanopoulos, R. Benedictus, T. Loutas, and D. Zarouchas, "Fusion-based damage diagnostics for stiffened composite panels," *Structural Health Monitoring*, vol. 21, no. 2, pp. 613–639, 2022.

[6] M. Saeedifar and D. Zarouchas, "Damage characterization of laminated composites using acoustic emission: A review," *Composites Part B: Engineering*, vol. 195, p. 108039, 2020.

[7] S. Arlot and A. Celisse, "A survey of cross-validation procedures for model selection," 2010.

[8] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.

[9] D. W. Marquardt, "An algorithm for least-squares estimation of nonlinear parameters," *Journal of the society for Industrial and Applied Mathematics*, vol. 11, no. 2, pp. 431–441, 1963.

[10] M. T. Hagan and M. B. Menhaj, "Training feedforward networks with the marquardt algorithm," *IEEE transactions on Neural Networks*, vol. 5, no. 6, pp. 989–993, 1994.

[11] M. T. Hagan, H. B. Demuth, and M. Beale, *Neural network design*. PWS Publishing Co., 1997.

[12] F. Foresee and M. Hagan, "Proceedings of the international joint conference on neural networks," 1997.

[13] D. J. MacKay, "Bayesian interpolation," *Neural computation*, vol. 4, no. 3, pp. 415–447, 1992.

[14] M. Riedmiller and H. Braun, "A direct adaptive method for faster backpropagation learning: The rprop algorithm," in *IEEE international conference on neural networks*, pp. 586–591, IEEE, 1993.

[15] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[16] P. J. Huber, "Robust estimation of a location parameter," in *Breakthroughs in statistics: Methodology and distribution*, pp. 492–518, Springer, 1992.

[17] R. Hartley and A. Zisserman, *Multiple view geometry in computer vision.* Cambridge university press, 2003.

[18] P. Wen, S. Zhao, S. Chen, and Y. Li, "A generalized remaining useful life prediction method for complex systems based on composite health indicator," *Reliability Engineering & System Safety*, vol. 205, p. 107241, 2021.

[19] N. Eleftheroglou, D. Zarouchas, T. Loutas, R. Alderliesten, and R. Benedictus, "Structural health monitoring data fusion for in-situ life prognosis of composite structures," *Reliability Engineering & System Safety*, vol. 178, pp. 40–54, 2018.

[20] D. Zarouchas and N. Eleftheroglou, "In-situ fatigue damage analysis and prognostics of composite structures based on health monitoring data," in *Fatigue Life Prediction of Composites and Composite Structures*, pp. 711–739, Elsevier, 2020.

[21] N. Eleftheroglou and T. Loutas, "Fatigue damage diagnostics and prognostics of composites utilizing structural health monitoring data and stochastic processes," *Structural Health Monitoring*, vol. 15, no. 4, pp. 473–488, 2016.

[22] N. Yue, A. Broer, W. Briand, M. Rébillat, T. Loutas, and D. Zarouchas, "Assessing stiffness degradation of stiffened composite panels in post-buckling compression-compression fatigue using guided waves," *Composite Structures*, vol. 293, p. 115751, 2022.

[23] K. Peng, R. Jiao, J. Dong, and Y. Pi, "A deep belief network based health indicator construction and remaining useful life prediction using improved particle filter," *Neurocomputing*, vol. 361, pp. 19–28, 2019.

[24] C. Wang, R. Xiong, J. Tian, J. Lu, and C. Zhang, "Rapid ultracapacitor life prediction with a convolutional neural network," *Applied Energy*, vol. 305, p. 117819, 2022.

[25] C.-G. Huang, X. Yin, H.-Z. Huang, and Y.-F. Li, "An enhanced deep learning-based fusion prognostic method for rul prediction," *IEEE Transactions on Reliability*, vol. 69, no. 3, pp. 1097–1109, 2019.

[26] Y. Jiang, C. Li, Z. Yang, Y. Zhao, and X. Wang, "Remaining useful life estimation combining two-step maximal information coefficient and temporal convolutional network with attention mechanism," *Ieee Access*, vol. 9, pp. 16323–16336, 2021.

[27] N. Eleftheroglou, D. Zarouchas, and R. Benedictus, "An adaptive probabilistic data-driven methodology for prognosis of the fatigue life of composite structures," *Composite Structures*, vol. 245, p. 112386, 2020.

[28] T. Loutas, N. Eleftheroglou, and D. Zarouchas, "A data-driven probabilistic framework towards the in-situ prognostics of fatigue life of composites based on acoustic emission data," *Composite Structures*, vol. 161, pp. 522–529, 2017.

6

Designing HI for T-Stiffener Composite Panels and Dogbone Specimens using Guided Waves

This chapter presents an AI-driven solution leveraging signal processing to design reliable HIs for composite structures without the need for historical data. Utilizing GW data, the methodology demonstrates high performance in HIs for T-single stiffener CFRP panels and dogbone CFRP specimens.

6.1 Introduction

Developing comprehensive HIs for composite structures encompassing various damage types is challenging due to the stochastic nature of damage accumulation and uncertain events (like impact) during operation. This complexity is amplified when striving for HIs independent of historical data. This chapter introduces an AI-driven framework, the Hilbert transform-convolutional neural network under the semi-supervised learning paradigm (HT-SSCNN), for designing reliable HIs. It exclusively utilizes current GW data, eliminating the need for historical information. EL techniques were also used to enhance HI quality while reducing deep learning randomness. The methodology is validated through investigations on T-single stiffener CFRP panels under compression-fatigue and dogbone CFRP specimens under tension-fatigue loadings, showing high performance of up to 93% and 81%, respectively, in prognostic criteria.

6.2 Experimental Campaigns

For the HT-SSCNN framework fed by GW data, two different datasets are investigated: ReMAP and NASA

ReMAP: The first dataset, ReMAP which already explained in Chapter 5, Section 5.2, contains five composite skin-stiffener panels that were subjected to C-C fatigue loading (see Figure 6.1(right) and Table 6.1). GW data collection is carried out with eight surface-attached PZT sensors, where four sensors are located on the skin panel, two sensors are located on top of the stiffener-skin bondline, and two sensors are attached on the stiffener web. The GW system operates with one PZT serving as the actuator with six excitation frequencies (50 kHz, 100 kHz, 125 kHz, 150 kHz, 200 kHz, and 250 kHz) and the remaining seven PZTs functioning as sensors, rotating through all eight PZTs.

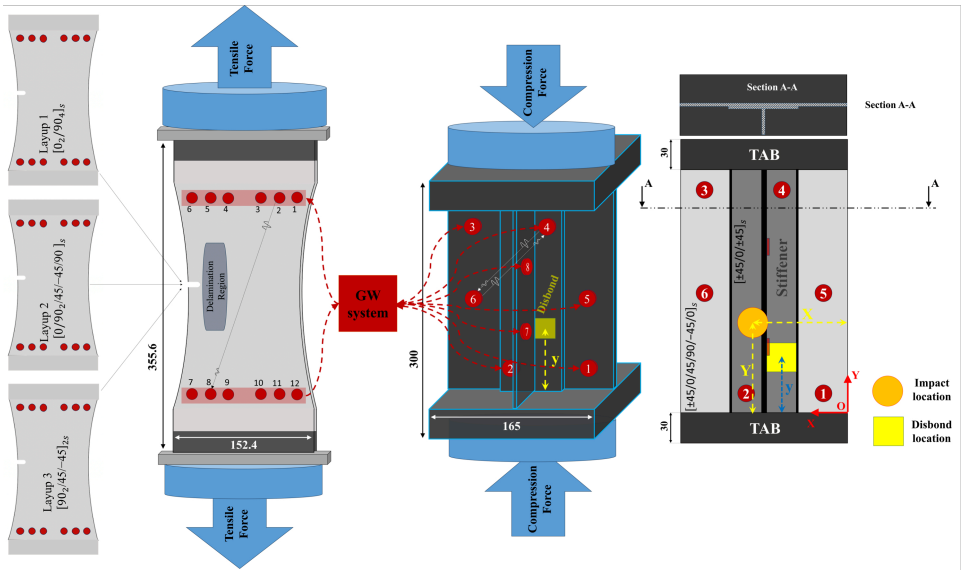


Figure 6.1: Composite dogbone specimens under T-T fatigue (left) and single T-stiffener CFRP panel under C-C fatigue (right) being monitored by PZT sensors (red circles) (dimensions in [mm]).

NASA: The second dataset, NASA, has three different layups, Layup1, Layup2, Layup3, with the ply orientation, $[0_2/90_4]_S$, $[0/90_2/\pm 45/90]_S$, and $[90_2/\pm 45]_{2S}$, respectively [1–3]. Torayca T700G unidirectional carbon-prepreg material has been used to manufacture the dogbone geometry coupons. The coupons with a notch have been submitted to tension-tension (T-T) fatigue load. GW acquisition has been performed through a surface-attached PZT network. The network contains one actuator and one sensor array with six linearly distributed PZT transducers in each, as shown in Figure 6.1(left). GW acquisition is realized between the linear arrays, which makes 36 actuator-sensor paths in total. Seven excitation frequencies have been applied in the range of 150 to 450 kHz with 50 kHz incrementation with a 5-cycle Hanning modulated tone-burst signal with an average input voltage of 50 V and a gain of 20 dB. Information regarding the mechanical test parameters can be found in Table 6.2. Further information regarding the mechanical experiment, GW data acquisition, etc. can be found in Refs. [1–3].

Table 6.1: Information of single T-stiffener CFRP panels tested under C-C fatigue loading (ReMAP dataset) monitored by guided waves.

Name	***L1-03***	***L1-04***	***L1-05***	***L1-09***	***L1-23***
Name index	SSP1	SSP2	SSP3	SSP4	SSP5
X-location of impact (mm)	50	25	115	82.5	-
Y-location of impact (mm)	80	80	160	140	-
Time of Impact	at 0 cycles	at 0 cycles	at 0 cycles	at 0 cycles	-
Size of disbond (mm)	-	-	-	-	30×30
y-location of disbond (mm)	-	-	-	-	60
Min Load (kN)	-6.5	-6.5	-6.5	-6.5	-5 and -6
Max Load (kN)	-65	-65	-65	-65	-50 and -60
Cycles	152,458	280,098	144,969	133,281	438,000

Table 6.2: Information of dogbone CFRP plates tested under T-T fatigue loading (NASA dataset) monitored by guided waves.

Ply orientation	***Layup 1:*** $[0_2/90_4]_S$				***Layup 2:*** $[0/90_2/\pm 45/90]_S$			***Layup 3:*** $[90_2/\pm 45]_{2S}$			
Name index	S1	S2	S3	S4	S1	S2	S3	S1	S2	S3	S4
Static Failure (Mpa)	133.8	133.8	133.8	133.8	147.5	147.5	147.5	104.8	104.8	104.8	104.8
Max Load (kN)	111.7	111.7	111.7	111.7	127.5	127.5	127.5	89.6	89.6	89.6	89.6
Load Ratio	0.14	0.14	0.14	0.14	0.14	0.14	0.14	0.13	0.13	0.13	0.13
Cycles (×1000)	227	100	1650	125	207	900	1250	150	300	895	7500

6.3 Framework: HT-SSCNN

A framework called HT-SSCNN is introduced to construct HIs fulfilling evaluation criteria based on historical-independent GW data. First, the GW signals recorded from the network of sensors and their envelopes are extracted using the Hilbert transform (HT). Then, these envelopes are integrated to reshape a 3D form input to feed the subsequent DL model. Convolutional neural network (CNN) architectures are designed to fuse the 3D inputs trained by a SSL paradigm, and their inherent randomness and uncertainty are mitigated by EL. All GW data generated from different triggering frequencies is also fused during the EL step. Six methods, including four average-based and two DL networks, are

considered for EL. In this section, the framework and step-by-step process are introduced as can be seen in Figure 6.2.

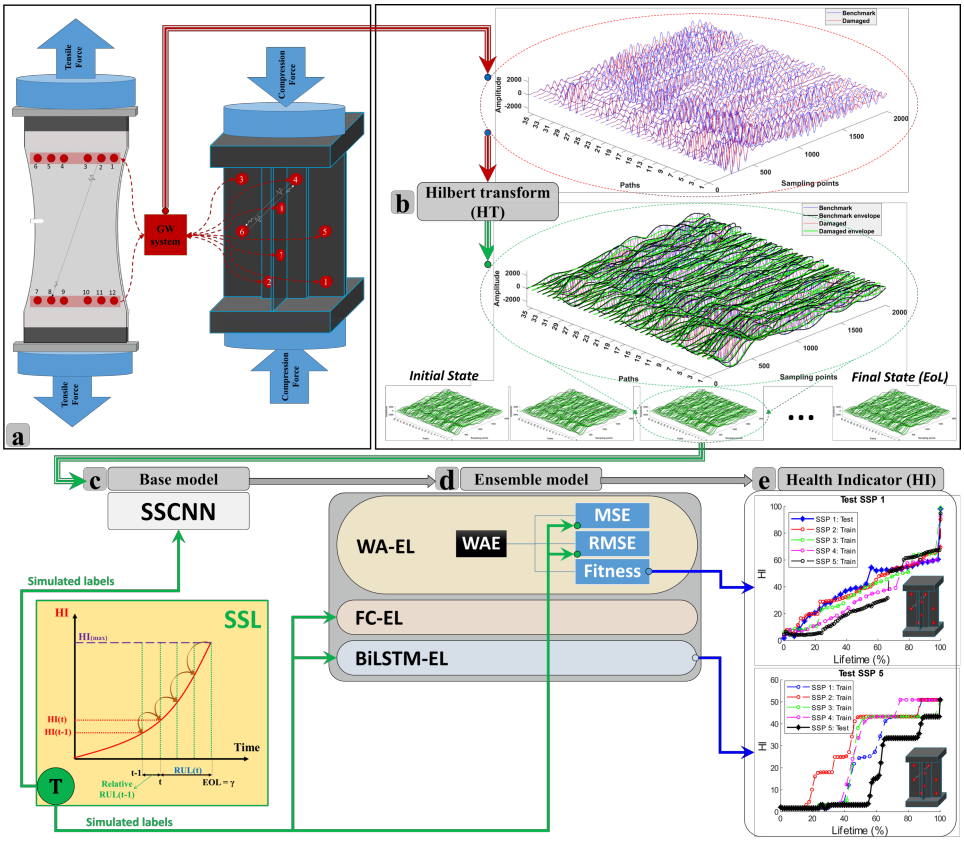


Figure 6.2: The overall proposed framework: (a) guided wave (GW) monitoring; (b) signal processing (Hilbert transform); (c) base learner model (SSCNN); (d) ensemble learner model;(e) extracted health indicator.

Signal processing and 3D input preparation: Prior to using DL networks, signal processing techniques with explicit and fast solutions can improve performance and simplify following DL modeling. An effective method involves extracting GW signal envelopes using the magnitude of their analytic signals, which is achieved through the HT (Chapter 4, Section 4.3.2). HT is replaced by a finite impulse response (FIR) filter in discrete-time signal processing to reduce computational complexity [4]. This specific FIR filter is termed the Hilbert transform FIR (HT-FIR) filter, with its length determined by the excitation frequency in the current work. For example, a 400 kHz frequency corresponds to an HT-FIR filter length of 400. The filter is created by applying a Kaiser window with a shape parameter $\beta = 8$ to an ideal brick-wall filter. Similar processing generates upper and lower envelopes for all GW signals.

A 3D form of [paths between actuators and sensors]×[signal length]×[upper and lower envelopes in two states—baseline and current time] is implemented to prepare inputs for the subsequent SSCNN model. Accordingly, the input dimensions are 36×2000×4 or

56×2000×4 for the NASA and ReMAP datasets, respectively. Figure 6.3 shows, for the NASA dataset (layup 1), the GW signals excited by one frequency (150 kHz), the extracted envelopes for all paths, and the pertinent 3D input of SSCNN at cycle 60000.

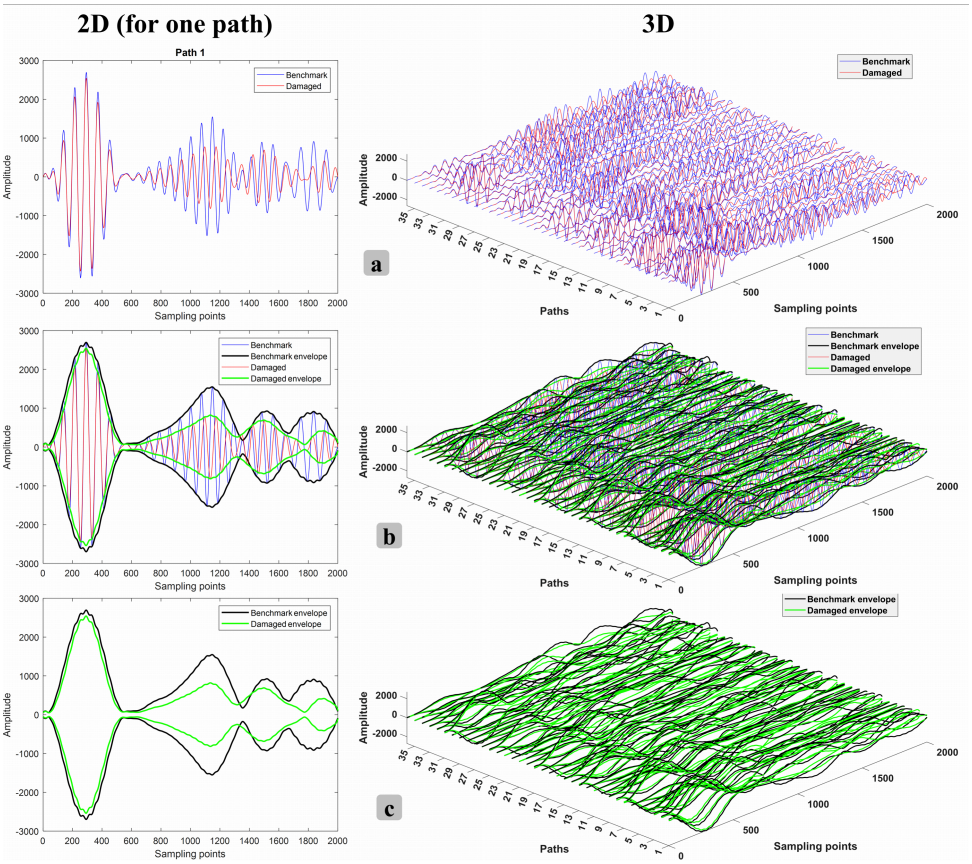


Figure 6.3: Right column: (a) Sensed GW signals excited by 150 kHz and (b) their envelopes for all 36 paths of the NASA dataset (layup 1) at baseline and cycle 60000, as well as the relevant (c) 3D input of SSCNN at cycle 60000. Left column: Their 2D display for only one path.

6.3.1 Feature fusion: semi-supervised convolutional neural network (SSCNN)

A CNN architecture, illustrated in Figure 6.4, has been designed to map GW inputs to the simulated ideal HI. The inputs, as previously described, take the 3D shape of $(N_a \times N_s) \times 2000 \times 4$, where N_a and N_s represent the number of actuators and sensors, respectively. The leakage coefficient for all Leaky ReLU functions is set at 0.01. To calculate the loss function between predictions and targets, an MSE is employed. An Adam optimizer was utilized to train the SSCNN with 38400 and 28800 learnable parameters for ReMAP and NASA setups, employing an initial learning rate of 0.001. SSCNN is considered the base learner model in the entire framework developed upon GW.

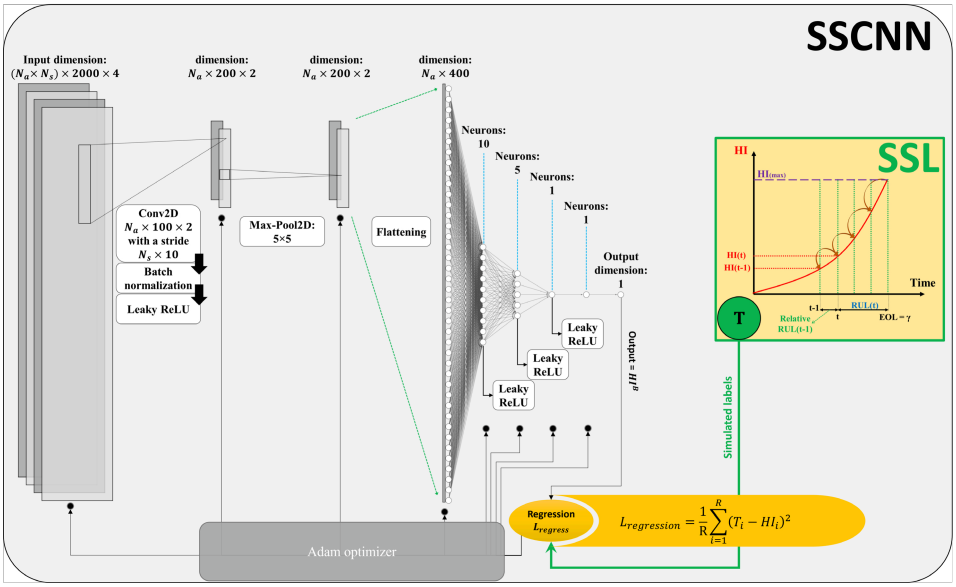


Figure 6.4: The architecture of the semi-supervised convolutional neural network (SSCNN) as the base learner to construct HIs using GW data.

6

Ensemble learning (EL)

Once HIs are constructed using the base learner model, ensemble learner models can handle uncertainties and randomness. The initial step involves LOOCV. Here, one unit (composite specimen) is set aside for testing, another for validation, and the rest are used for training. The validation unit rotates through all available options, and the base learner model (SSCNN) is trained ten times with different random seed numbers for weight and bias initialization.

Beyond averaging ensemble models, different networks with various layer types were explored, among which the two best ones are presented. The DL architectures are summarized in Table 6.3, with the number of neurons, units, or dropout (D) percentages indicated in parentheses.

Table 6.3: Ensemble learner models used on top of SSCNN.

Model num.	*Model name*	*Architecture (hidden layers)*						
1	**SAE**							
2	**WAE-MSE**							
3	**WAE-RMSE**							
4	**WAE-Fitness**							
5	**FC-Net**	FC(100)	D(0.5)	ReLU	FC(5)	D(0.5)	ReLU	FC(1)
6	**BiLSTM-Net**	D(0.5)	BiLSTM(5)	D(0.5)	FC(5)	D(0.5)	ReLU	FC(1)

6.3.2 Results

To thoroughly evaluate various combinations and confirm the effectiveness, validity, and stability of the proposed framework, all potential dataset folds were examined. In each fold, one specimen served as the test set, while the remaining composite specimens were used for training and validation. For both the ReMAP and NASA datasets, which offered four and three-two alternatives for validation, respectively, the process was repeated with 10 iterations using distinct random seed numbers. This approach essentially mirrors the LOOCV, employing a holdout validation strategy within each fold to assess the model's generalizability. In this section, the results of the HT-SSCNN up to the ensemble stage, referred to as the base learner models, are initially presented before delving into the outcomes of the ensemble learner models.

Base learner models

The comprehensive results of HT-SSCNN across various subsets and excitation frequencies for ReMAP dataset are presented in Figure 6.5. The results display fitness scores (based on Eq. 3.4) across different replications and their mean values (indicated on the right y-axis) over ten replications (shown on the x-axis) for each subset (validation index-based combinations displayed on the left y-axis), with error bars representing the standard deviation. The impact of the choice of the validation unit varies depending on the specific test unit and frequency. For instance, in fold 5, where the test unit is specimen 5, unit 1 does not serve as a suitable validation case for GW signals at the excitation frequency of 50 kHz. This is either because it leads to the exclusion of unit 1 information during the training phase or it proves to be an unsuitable validation specimen for terminating the training process. Referring to Figure 6.5, it becomes evident that frequencies of 150 kHz and 200 kHz led to more consistent high fitness scores, whereas 100 kHz and 250 kHz exhibit unstable performance. Appendices (A.3) also include analogous illustrations for the NASA dataset, highlighting the influence of validation unit selection and GW excitation frequency. Tables 6.4, 6.5, 6.6, and 6.7 present averaged fitness values (±standard deviation) across all repetitions, irrespective of the validation unit chosen, for various folds. F-All corresponds to the fitness scores obtained from Eq. 3.4 considering all units, while F-Test relates to the fitness scores obtained from Eq. 3.8 focusing solely on the test unit.

The highest-scoring frequency is 150 kHz for the ReMAP dataset, with an average of 2.53 for all units and 2.26 for test units across all folds. In the NASA datasets, the top-performing frequencies are 250 kHz, 300 kHz, and 350 kHz for layups 1, 2, and 3, respectively. The average scores across all folds are 2.37, 2.19, and 2.32 when considering all units, and 2.04, 1.97, and 2.04 when considering test units, as highlighted in green in the tables. Layup 2 yields lower scores because it has limited training data from only one unit.

The results vary across different folds, with the folds showing the highest fitness scores highlighted in bold in the tables. The HIs generated from HT-SSCNN for the optimal frequencies in different datasets are displayed in Figure 6.6. These HIs represent the averages across all repetitions, irrespective of the validation unit selection (SAE method). However, showcasing the HIs from a single repetition could yield more promising results.

For T-single stiffener CFRP panels, the results are highly promising despite the presence of uncertainties such as broken sensors, impacts, and disbond. In the NASA dataset,

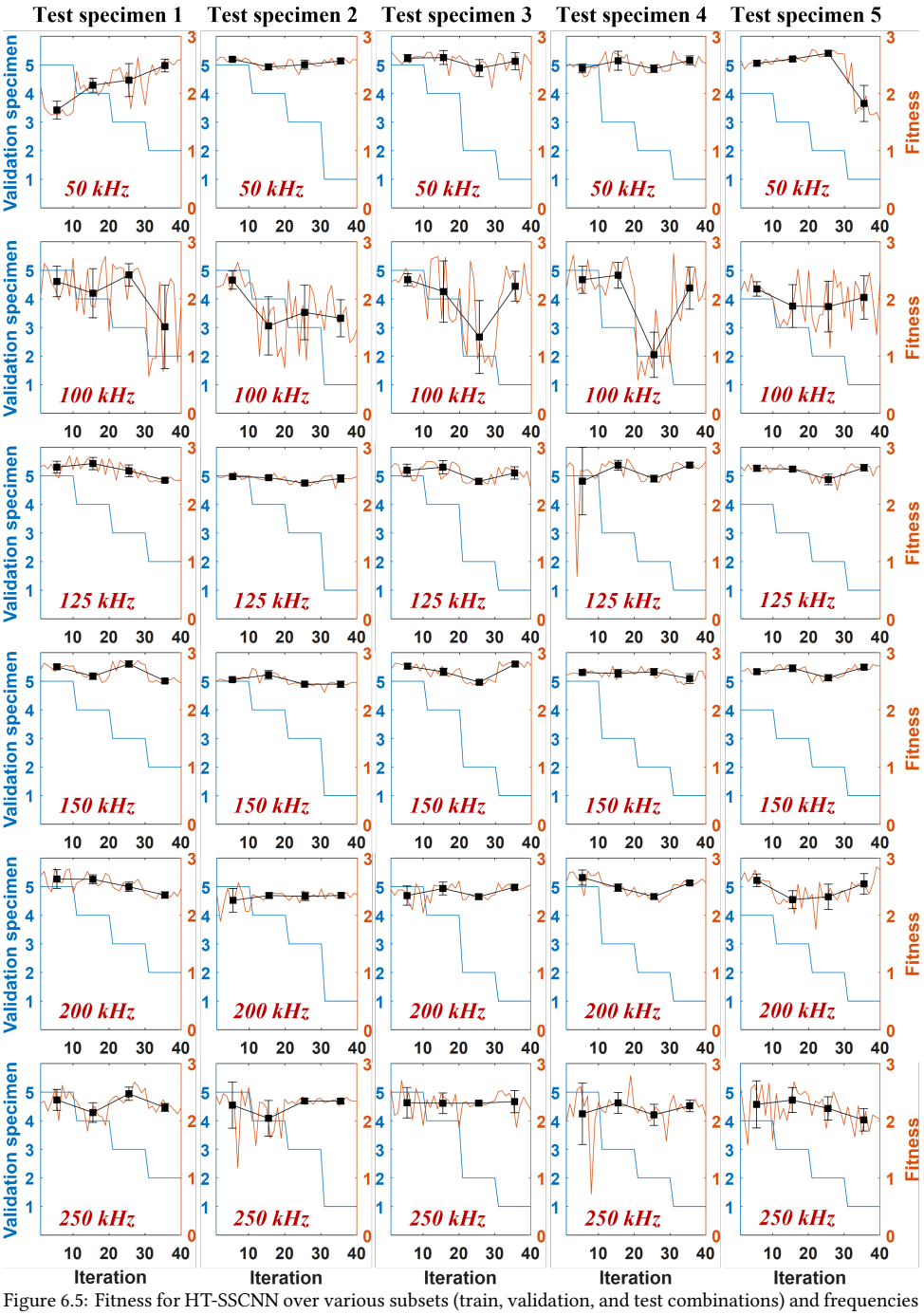


Figure 6.5: Fitness for HT-SSCNN over various subsets (train, validation, and test combinations) and frequencies in the ReMAP dataset (fitness calculated from Eq. 3.4, considering all units referred to as 'F-All'). Black lines illustrate the variance (Std) bars of the fitness with filled square markers denoting its mean value, plotted at every 10 iterations.

Table 6.4: Fitness values for base learner models averaged over the 40 repetitions for ReMAP dataset.

Frequency	*Fold 1*		*Fold 2*		*Fold 3*		*Fold 4*		*Fold 5*	
	F-All	F-Test	F-All	F-Test	F-All	F-Test	F-All	F-Test	F-All	F-Test
50 kHz	2.23 (±0.15)	1.55 (±0.34)	2.31 (±0.37)	2.12 (±0.45)	2.46 (±0.25)	2.22 (±0.48)	2.41 (±0.42)	2.17 (±0.35)	2.49 (±0.5)	2.28 (±0.5)
100 kHz	2.44 (±0.35)	2.25 (±0.49)	2.03 (±0.26)	1.55 (±0.3)	2.5 (±0.23)	2.23 (±0.23)	2.57 (±0.23)	2.41 (±0.38)	2.28 (±0.33)	2 (±0.37)
125 kHz	2.54 (±0.31)	2.4 (±0.44)	2.21 (±0.29)	1.93 (±0.24)	2.52 (±0.21)	2.11 (±0.35)	2.55 (±0.33)	2.36 (±0.43)	2.55 (±0.2)	2.21 (±0.31)
150 kHz	2.59 (±0.45)	2.43 (±0.48)	2.19 (±0.34)	2.01 (±0.44)	**2.72 (±0.14)**	**2.43 (±0.3)**	2.54 (±0.19)	2.14 (±0.33)	2.6 (±0.22)	2.3 (±0.29)
200 kHz	2.45 (±0.3)	2.21 (±0.45)	2.16 (±0.31)	1.91 (±0.33)	2.5 (±0.09)	2.09 (±0.24)	2.63 (±0.14)	2.44 (±0.46)	**2.62 (±0.32)**	**2.55 (±0.39)**
250 kHz	2.51 (±0.51)	2.38 (±0.52)	2.21 (±0.26)	1.93 (±0.24)	2.35 (±0.2)	1.96 (±0.35)	2.29 (±0.4)	2.09 (±0.55)	2.45 (±0.25)	1.89 (±0.21)
All	2.55 (±0.32)	2.4 (±0.46)	2.19 (±0.37)	1.84 (±0.38)	2.62 (±0.2)	2.42 (±0.3)	2.64 (±0.25)	2.48 (±0.37)	2.53 (±0.49)	2.33 (±0.55)

Table 6.5: Fitness values for base learner models averaged over the 30 repetitions for NASA dataset (layup 1).

Frequency	*Fold 1*		*Fold 2*		*Fold 3*		*Fold 4*	
	F-All	F-Test	F-All	F-Test	F-All	F-Test	F-All	F-Test
150 kHz	1.86 (±0.36)	1.16 (±0.47)	2.18 (±0.42)	2.08 (±0.47)	1.67 (±0.29)	1.21 (±0.38)	2.32 (±0.12)	2.18 (±0.16)
200 kHz	1.99 (±0.4)	1.4 (±0.49)	2.49 (±0.32)	2.21 (±0.36)	1.52 (±0.32)	0.86 (±0.51)	2.4 (±0.22)	2.12 (±0.45)
250 kHz	2.41 (±0.24)	2.03 (±0.31)	2.3 (±0.35)	1.87 (±0.32)	2.22 (±0.16)	1.92 (±0.22)	**2.54 (±0.26)**	**2.32 (±0.32)**
300 kHz	2.47 (±0.2)	2.11 (±0.24)	2.56 (±0.15)	2.14 (±0.27)	1.98 (±0.22)	1.61 (±0.28)	2.52 (±0.2)	2.13 (±0.33)
350 kHz	2.12 (±0.35)	1.71 (±0.39)	1.9 (±0.39)	1.43 (±0.52)	1.84 (±0.25)	1.45 (±0.25)	2.11 (±0.41)	1.57 (±0.75)
400 kHz	2.02 (±0.29)	1.52 (±0.58)	1.85 (±0.46)	1.24 (±0.51)	1.73 (±0.28)	1.18 (±0.42)	2.12 (±0.34)	1.91 (±0.53)
450 kHz	2.35 (±0.38)	2.05 (±0.44)	2 (±0.42)	1.59 (±0.55)	1.71 (±0.36)	1.25 (±0.47)	2.26 (±0.35)	1.87 (±0.64)
All	2.23 (±0.35)	1.86 (±0.48)	2.36 (±0.3)	2.06 (±0.28)	2.07 (±0.4)	1.65 (±0.43)	2.37 (±0.32)	2.28 (±0.48)

Table 6.6: Fitness values for base learner models averaged over the 20 repetitions for NASA dataset (layup 2).

Frequency	*Fold 1*		*Fold 2*		*Fold 3*	
	F-All	F-Test	F-All	F-Test	F-All	F-Test
150 kHz	1.47 (±0.56)	1.11 (±0.76)	1.72 (±0.39)	1.36 (±0.65)	1.6 (±0.24)	1.37 (±0.24)
200 kHz	1.92 (±0.54)	1.55 (±0.83)	1.84 (±0.48)	1.64 (±0.55)	1.8 (±0.33)	1.54 (±0.27)
250 kHz	1.67 (±0.45)	1.47 (±0.54)	2.02 (±0.46)	1.75 (±0.62)	1.89 (±0.3)	1.69 (±0.4)
300 kHz	2.1 (±0.34)	1.96 (±0.43)	2.21 (±0.34)	1.79 (±0.55)	**2.27 (±0.26)**	**2.17 (±0.29)**
350 kHz	1.6 (±0.54)	1.37 (±0.62)	1.78 (±0.52)	1.42 (±0.62)	1.89 (±0.34)	1.7 (±0.27)
400 kHz	1.6 (±0.42)	1.14 (±0.57)	1.43 (±0.35)	0.92 (±0.44)	1.68 (±0.45)	1.53 (±0.49)
450 kHz	1.75 (±0.45)	1.46 (±0.64)	1.6 (±0.4)	1.16 (±0.5)	1.91 (±0.3)	1.64 (±0.31)
All	2.05 (±0.58)	1.91 (±0.59)	2.07 (±0.53)	1.79 (±0.68)	2.08 (±0.46)	1.98 (±0.35)

Table 6.7: Fitness values for base learner models averaged over the 30 repetitions for NASA dataset (layup 3).

Frequency	*Fold 1*		*Fold 2*		*Fold 3*		*Fold 4*	
	F-All	F-Test	F-All	F-Test	F-All	F-Test	F-All	F-Test
150 kHz	1.78 (±0.42)	1.22 (±0.64)	2.28 (±0.23)	2.13 (±0.4)	2.32 (±0.19)	1.97 (±0.23)	1.83 (±0.34)	1.26 (±0.46)
200 kHz	2.37 (±0.33)	2.1 (±0.55)	2.46 (±0.31)	2.43 (±0.4)	2.6 (±0.14)	2.31 (±0.25)	1.83 (±0.32)	1.29 (±0.39)
250 kHz	2.37 (±0.33)	2.16 (±0.6)	2.59 (±0.19)	2.5 (±0.14)	2.25 (±0.27)	1.89 (±0.44)	1.91 (±0.39)	1.45 (±0.53)
300 kHz	2.27 (±0.34)	1.91 (±0.55)	2.47 (±0.44)	2.33 (±0.57)	2.44 (±0.22)	2.18 (±0.4)	1.87 (±0.34)	1.33 (±0.45)
350 kHz	2.46 (±0.25)	2.24 (±0.4)	2.28 (±0.33)	2.06 (±0.45)	**2.55 (±0.21)**	**2.38 (±0.3)**	1.99 (±0.36)	1.49 (±0.49)
400 kHz	2.31 (±0.39)	1.98 (±0.57)	2.28 (±0.24)	1.93 (±0.45)	2.52 (±0.18)	2.31 (±0.24)	1.91 (±0.35)	1.46 (±0.49)
450 kHz	2.5 (±0.2)	2.23 (±0.32)	2.25 (±0.44)	2.02 (±0.63)	2.57 (±0.18)	2.22 (±0.26)	1.98 (±0.37)	1.51 (±0.51)
All	2.22 (±0.47)	2.03 (±0.56)	2.54 (±0.17)	2.41 (±0.22)	2.46 (±0.34)	2.25 (±0.34)	1.82 (±0.33)	1.24 (±0.37)

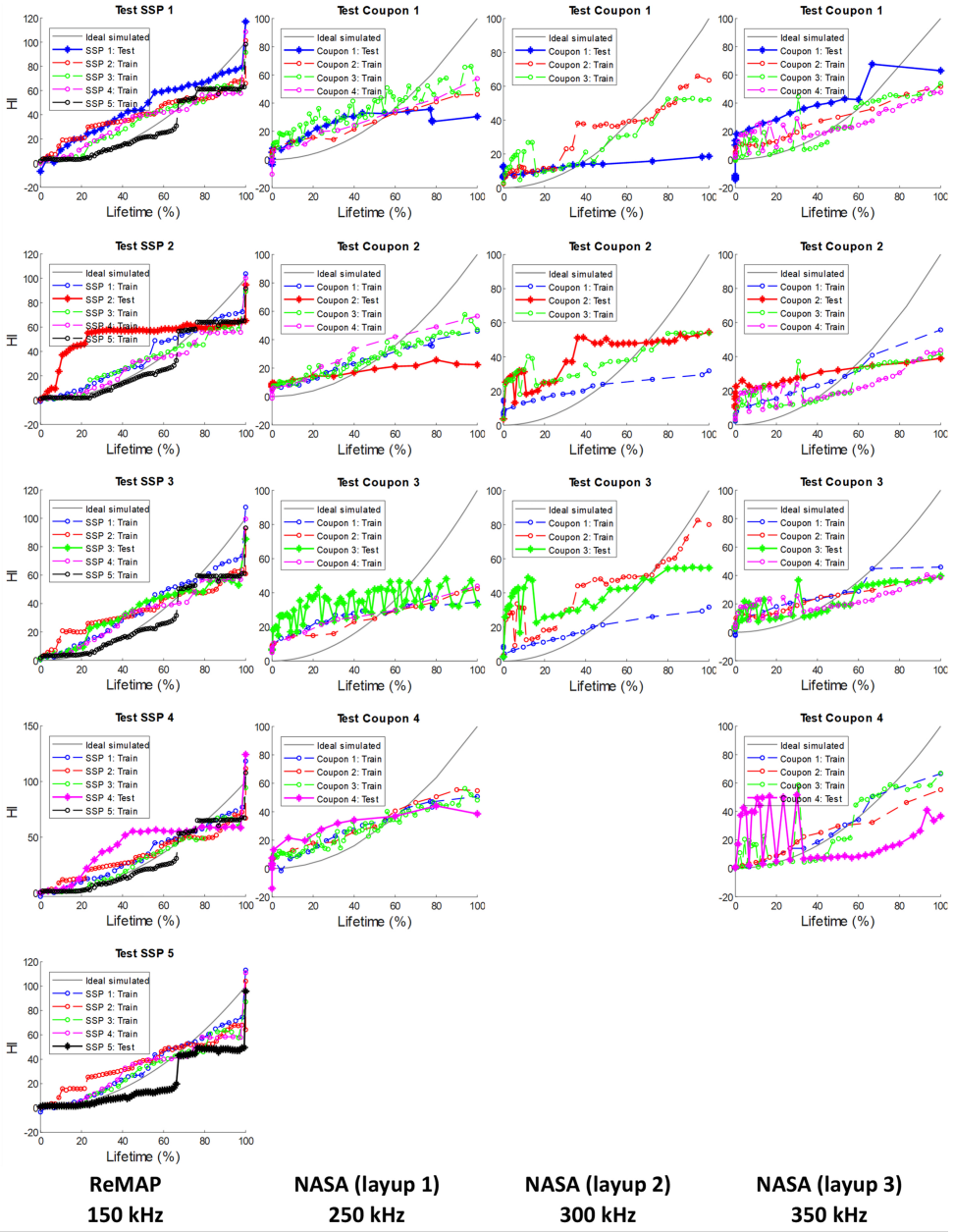


Figure 6.6: HIs obtained by the proposed framework (with SAE) for different datasets, given single frequency input. The actuating GW frequencies were selected based on the best fitness scores (Tables 6.4, 6.5, 6.6, and 6.7).

Layup 2 is hindered by a lack of training data, while the HIs for Layup 1 (except for unit 2) and Layup 3 (except for units 1 and 4) outperform others.

Ensemble learner models

Tables 6.8, 6.9, 6.10, and 6.11 present the ensemble models' fitness scores averaged across all folds for various ensemble techniques. EL models can be applied to the HIs generated by HT-SSCNN using a single excitation GW frequency or a fusion of all frequencies (labeled 'Fused (all)' in Tables 6.8-6.11).

For the ReMAP dataset, the WAE-Fitness model using the fusion of all GW excitation frequencies achieved the best fitness scores whether considering all or only test units, with F-All of 2.78 and F-Test of 2.67. Additionally, SAE and the FC network (FC-EL) models using the fusion of frequencies also resulted in high scores. Aside from the fusion of all frequencies, the 150 kHz frequency, when employed with SAE and WAE-Fitness, yielded high fitness scores (exceeding 2.6, meeting 87% of evaluation criteria).

In the NASA dataset, for Layup 1, the BiLSTM network (BiLSTM-EL using the fusion of all frequencies obtained the highest fitness scores (F-All of 2.68 and F-Test of 2.31). However, it is important to note that the HI construction model for Layup 1 is not historical-independent as it uses prior information in the EL step. WAE-Fitness with GW signals generated from the 250 kHz frequency achieved better fitness scores in second place. The WAE-Fitness using the fused frequencies for this layup generates acceptable scores (F-All of 2.41 and F-Test of 2.03). For Layup 2, EL models on the GW excitation frequency of 300 kHz provided higher scores than others, but the fitness scores are not yet highly proper as this dataset, as previously mentioned, suffers from the limited data needed to train the models. In Layup 3, WAE-Fitness on the fusion of all frequencies achieved the highest fitness scores (F-All of 2.42 and F-Test of 2.23).

Table 6.8: Fitness values for ensemble learner models averaged over the 5 Folds for ReMAP dataset.

Freq. (kHz)	*SAE*		*WAE*						*Deep learning*			
			MSE		*RMSE*		*Fitness*		*FC*		*BiLSTM*	
	F-All	F-Test	F-All	F-Test	F-All	F-Test	F-All	F-Test	F-All	F-Test	F-All	F-Test
50	2.58 (±0.24)	2.32 (±0.6)	2.58 (±0.21)	2.33 (±0.56)	2.58 (±0.23)	2.32 (±0.59)	2.58 (±0.24)	2.32 (±0.6)	2.6 (±0.16)	2.29 (±0.44)	2 (±0.22)	1.64 (±0.16)
100	2.45 (±0.26)	2.13 (±0.52)	2.25 (±0.44)	2.04 (±0.53)	2.34 (±0.37)	2.08 (±0.51)	2.46 (±0.26)	2.12 (±0.52)	2.62 (±0.26)	2.33 (±0.47)	2.31 (±0.34)	1.97 (±0.52)
125	2.69 (±0.12)	2.52 (±0.26)	2.66 (±0.12)	2.5 (±0.26)	2.68 (±0.12)	2.51 (±0.27)	2.69 (±0.12)	2.52 (±0.27)	2.68 (±0.13)	2.5 (±0.22)	2.24 (±0.22)	1.75 (±0.2)
150	**2.73 (±0.12)**	**2.6 (±0.11)**	2.7 (±0.11)	2.58 (±0.09)	2.72 (±0.12)	2.59 (±0.1)	**2.73 (±0.12)**	**2.6 (±0.11)**	2.71 (±0.14)	2.45 (±0.35)	2.43 (±0.39)	2 (±0.36)
200	2.6 (±0.14)	2.51 (±0.38)	2.57 (±0.16)	2.47 (±0.41)	2.57 (±0.17)	2.48 (±0.41)	2.6 (±0.14)	2.51 (±0.38)	2.6 (±0.15)	2.49 (±0.37)	2.49 (±0.24)	2.18 (±0.52)
250	2.54 (±0.14)	2.25 (±0.35)	2.49 (±0.12)	2.27 (±0.31)	2.52 (±0.13)	2.26 (±0.34)	2.54 (±0.14)	2.25 (±0.32)	2.49 (±0.18)	2.1 (±0.32)	2.59 (±0.12)	2.26 (±0.46)
Fused (all)	**2.77 (±0.15)**	**2.66 (±0.22)**	2.71 (±0.23)	2.57 (±0.4)	2.75 (±0.18)	2.64 (±0.26)	**2.78 (±0.15)**	**2.67 (±0.2)**	**2.76 (±0.14)**	**2.62 (±0.22)**	2.38 (±0.37)	2.01 (±0.43)

On average, WAE-Fitness using the fusion of all GW excitation frequencies resulted in higher fitness scores, and the related constructed HIs using this model (i.e., the end-to-end model made of HT-SSCNN-WAE$_{Fitness}$) can be seen in Figure 6.7. Considering the scores reported in Table 8 and the generated HIs in Figure 6.7 for T-single CFRP panels under C-C fatigue loading, the performance is highly satisfactory, achieving 93% (2.78 / 3.00) given all units and 89% (2.67 / 3.00) given test units. The HIs exhibit monotonic, prognostic, and trendable behavior, making them suitable for predicting the RUL of composite structures.

For dogbone CFRP specimens under T-T fatigue loading, layups with four units (Layup

Table 6.9: Fitness values for ensemble learner models averaged over the 4 Folds for NASA dataset (layup 1).

Freq. (kHz)	*SAE*		*WAE*						*Deep learning*			
			MSE		*RMSE*		*Fitness*		*FC*		*BiLSTM*	
	F-All	F-Test	F-All	F-Test	F-All	F-Test	F-All	F-Test	F-All	F-Test	F-All	F-Test
150	2.14 (±0.37)	1.71 (±0.59	2.05 (±0.28)	1.73 (±0.51)	2.09 (±0.31)	1.73 (±0.53)	2.15 (±0.37)	1.72 (±0.59)	1.78 (±0.54)	1.43 (±0.92)	2.23 (±0.41)	1.87 (±0.23)
200	2.24 (±0.52)	1.8 (±0.71)	2.22 (±0.47)	1.86 (±0.71)	2.23 (±0.48)	1.85 (±0.7)	2.25 (±0.52)	1.79 (±0.7)	2.12 (±0.53)	1.54 (±0.9)	2.14 (±0.56)	1.77 (±0.91)
250	**2.52 (±0.23)**	**2.12 (±0.26)**	**2.5 (±0.22)**	**2.12 (±0.26)**	**2.5 (±0.23)**	**2.12 (±0.25)**	**2.52 (±0.21)**	**2.13 (±0.23)**	2.45 (±0.24)	2.1 (±0.39)	2.33 (±0.21)	1.89 (±0.33)
300	2.47 (±0.36)	2.06 (±0.32)	2.41 (±0.35)	2.06 (±0.36)	2.44 (±0.35)	2.07 (±0.33)	2.48 (±0.36)	2.08 (±0.32)	2.36 (±0.37)	1.96 (±0.33)	2.34 (±0.44)	1.84 (±0.52)
350	2.19 (±0.41)	1.82 (±0.52)	2.16 (±0.38)	1.85 (±0.51)	2.18 (±0.39)	1.85 (±0.49)	2.2 (±0.4)	1.82 (±0.52)	1.93 (±0.44)	1.26 (±0.51)	1.78 (±0.42)	1.24 (±0.41)
400	2.11 (±0.38)	1.7 (±0.52)	2.06 (±0.32)	1.73 (±0.46)	2.09 (±0.35)	1.73 (±0.48)	2.11 (±0.36)	1.69 (±0.51)	1.89 (±0.41)	1.26 (±0.75)	1.73 (±0.18)	1.18 (±0.31)
450	2.37 (±0.45)	1.95 (±0.62)	2.33 (±0.45)	1.99 (±0.68)	2.37 (±0.45)	2 (±0.68)	2.36 (±0.45)	1.93 (±0.57)	2.01 (±0.56)	1.58 (±0.63)	2.04 (±0.46)	1.73 (±0.74)
Fused (all)	2.39 (±0.44)	2.02 (±0.64)	2.35 (±0.39)	2.01 (±0.57)	2.36 (±0.41)	2.03 (±0.61)	**2.41 (±0.43)**	**2.03 (±0.63)**	2.29 (±0.5)	1.92 (±0.6)	**2.68 (±0.2)**	**2.31 (±0.36)**

Table 6.10: Fitness values for ensemble learner models averaged over the 3 Folds for NASA dataset (layup 2).

Freq. (kHz)	*SAE*		*WAE*						*Deep learning*			
			MSE		*RMSE*		*Fitness*		*FC*		*BiLSTM*	
	F-All	F-Test	F-All	F-Test	F-All	F-Test	F-All	F-Test	F-All	F-Test	F-All	F-Test
150	1.62 (±0.55)	1.24 (±0.87)	1.48 (±0.29)	1.19 (±0.68)	1.63 (±0.51)	1.3 (±0.89)	1.67 (±0.56)	1.38 (±0.98)	1.43 (±0.1)	0.9 (±0.52)	1.58 (±0.45)	1.3 (±0.39)
200	1.6 (±0.37)	1.33 (±0.8)	1.57 (±0.23)	1.29 (±0.64)	1.57 (±0.3)	1.26 (±0.74)	1.58 (±0.29)	1.35 (±0.6)	1.42 (±0.28)	0.84 (±0.51)	2.07 (±0.09)	1.72 (±0.15)
250	2.25 (±0.25)	2.12 (±0.42)	2.28 (±0.28)	2.15 (±0.33)	2.27 (±0.27)	2.16 (±0.39)	2.25 (±0.26)	2.11 (±0.47)	1.97 (±0.21)	1.73 (±0.35)	1.94 (±0.39)	1.79 (±0.47)
300	**2.31 (±0.12)**	**2.25 (±0.14)**	**2.33 (±0.08)**	**2.25 (±0.13)**	**2.32 (±0.1)**	**2.27 (±0.16)**	**2.3 (±0.12)**	**2.25 (±0.14)**	2.08 (±0.24)	1.7 (±0.54)	1.67 (±0.28)	1.15 (±0.46)
350	1.79 (±0.31)	1.55 (±0.24)	1.82 (±0.31)	1.62 (±0.28)	1.81 (±0.31)	1.59 (±0.26)	1.8 (±0.34)	1.53 (±0.33)	1.3 (±0.31)	0.95 (±0.41)	1.8 (±0.77)	1.68 (±0.64)
400	1.58 (±0.17)	1.08 (±0.5)	1.61 (±0.13)	1.17 (±0.41)	1.62 (±0.14)	1.19 (±0.42)	1.69 (±0.23)	1.21 (±0.51)	1.32 (±0.23)	1.01 (±0.47)	1.54 (±0.79)	1.32 (±0.94)
450	1.73 (±0.32)	1.33 (±0.43)	1.72 (±0.26)	1.44 (±0.37)	1.76 (±0.33)	1.39 (±0.38)	1.76 (±0.35)	1.39 (±0.41)	1.27 (±0.29)	0.91 (±0.52)	1.79 (±0.35)	1.67 (±0.46)
Fused (all)	1.87 (±0.43)	1.75 (±0.53)	1.76 (±0.43)	1.61 (±0.56)	1.84 (±0.44)	1.72 (±0.6)	1.92 (±0.4)	1.83 (±0.53)	1.46 (±0.59)	1.66 (±0.37)	1.53 (±0.76)	1.5 (±0.83)

1 and Layup 3), which have 2 units for training and 1 unit for validation, yield better results compared to Layup 2, which has only one unit for training and another for validation. This can be observed in Figure 6.7, particularly for coupon 4 of Layup 1 and coupons 1, 2, and 3 of Layup 3.

HIs constructed by BiLSTM-EL upon HT-SSCNN are displayed in Figure 6.8. These HIs interestingly exhibit multiple incremental steps over the fatigue life in the ReMAP dataset and Layup 1 of the NASA dataset. These steps may signify distinct damage states, providing valuable insights for subsequent prognostic models, especially state-based ones, for RUL prediction. However, establishing a meaningful link between these steps and physical damage states requires further research and experimentation. The HIs constructed by FC-EL after the HT-SSCNN model can be found in Appendices (A.3).

In general, higher fitness scores for HIs could have been attained for the ReMAP

Table 6.11: Fitness values for Ensemble learner models averaged over the 4 Folds for NASA dataset (layup 3).

Freq. (kHz)	*SAE*		*WAE*						*Deep learning*			
			MSE		*RMSE*		*Fitness*		*FC*		*BiLSTM*	
	F-All	F-Test	F-All	F-Test	F-All	F-Test	F-All	F-Test	F-All	F-Test	F-All	F-Test
150	2.23 (±0.45)	1.96 (±0.62)	2.21 (±0.43)	1.97 (±0.62)	2.21 (±0.44)	1.97 (±0.62)	2.23 (±0.45)	1.94 (±0.62)	1.96 (±0.41)	1.42 (±0.66)	2.17 (±0.41)	1.73 (±0.57)
200	2.35 (±0.53)	2.22 (±0.82)	2.28 (±0.48)	2.17 (±0.78)	2.3 (±0.49)	2.18 (±0.79)	2.36 (±0.54)	2.23 (±0.82)	2.37 (±0.52)	2.14 (±0.83)	2.32 (±0.47)	1.91 (±0.39)
250	2.26 (±0.45)	2.16 (±0.75)	2.11 (±0.36)	2.05 (±0.66)	2.18 (±0.41)	2.1 (±0.7)	2.3 (±0.48)	2.18 (±0.77)	2.32 (±0.56)	1.99 (±0.73)	2.37 (±0.25)	1.93 (±0.54)
300	2.38 (±0.54)	2.2 (±0.82)	2.18 (±0.42)	2.08 (±0.74)	2.26 (±0.47)	2.13 (±0.77)	2.41 (±0.56)	2.22 (±0.83)	2.33 (±0.56)	2 (±0.82)	2.22 (±0.35)	1.79 (±0.6)
350	2.4 (±0.52)	2.2 (±0.75)	2.24 (±0.42)	2.09 (±0.68)	2.32 (±0.47)	2.15 (±0.71)	2.41 (±0.53)	2.21 (±0.76)	2.41 (±0.5)	2.08 (±0.71)	2.3 (±0.5)	2.14 (±0.54)
400	2.25 (±0.47)	2.05 (±0.68)	2.19 (±0.42)	2.06 (±0.68)	2.2 (±0.43)	2.03 (±0.66)	2.28 (±0.48)	2.07 (±0.69)	2.28 (±0.44)	1.85 (±0.64)	2.1 (±0.5)	1.69 (±0.67)
450	2.36 (±0.48)	2.15 (±0.72)	2.3 (±0.44)	2.11 (±0.69)	2.32 (±0.46)	2.12 (±0.7)	2.37 (±0.5)	2.16 (±0.73)	2.39 (±0.52)	2.09 (±0.72)	2.27 (±0.41)	1.83 (±0.56)
Fused (all)	**2.4 (±0.52)**	**2.22 (±0.77)**	2.29 (±0.46)	2.17 (±0.75)	2.33 (±0.48)	2.2 (±0.76)	**2.42 (±0.54)**	**2.23 (±0.78)**	**2.43 (±0.59)**	**2.16 (±0.82)**	2.09 (±0.45)	1.58 (±0.63)

datasets compared to the NASA dataset. This can be attributed to several factors, including the larger number of training specimens (only one unit more) and a greater number of time steps for GW inspections, which provide more training data for the DL models. This difference in performance seems to be less related to the structural type, loading conditions, layups, or types of damage, as the T-single stiffener panels are inherently more complex in various aspects. Moreover, the ReMAP structures were monitored using a more intertwined GW network with 56 paths, while the NASA structures used 36 paths. This richer, intertwined GW network offers more information to leverage for model training. In essence, the model's performance is less affected by issues encountered during the monitoring process, such as the presence of broken or debonded PZT sensors, which occurred during ReMAP experiments.

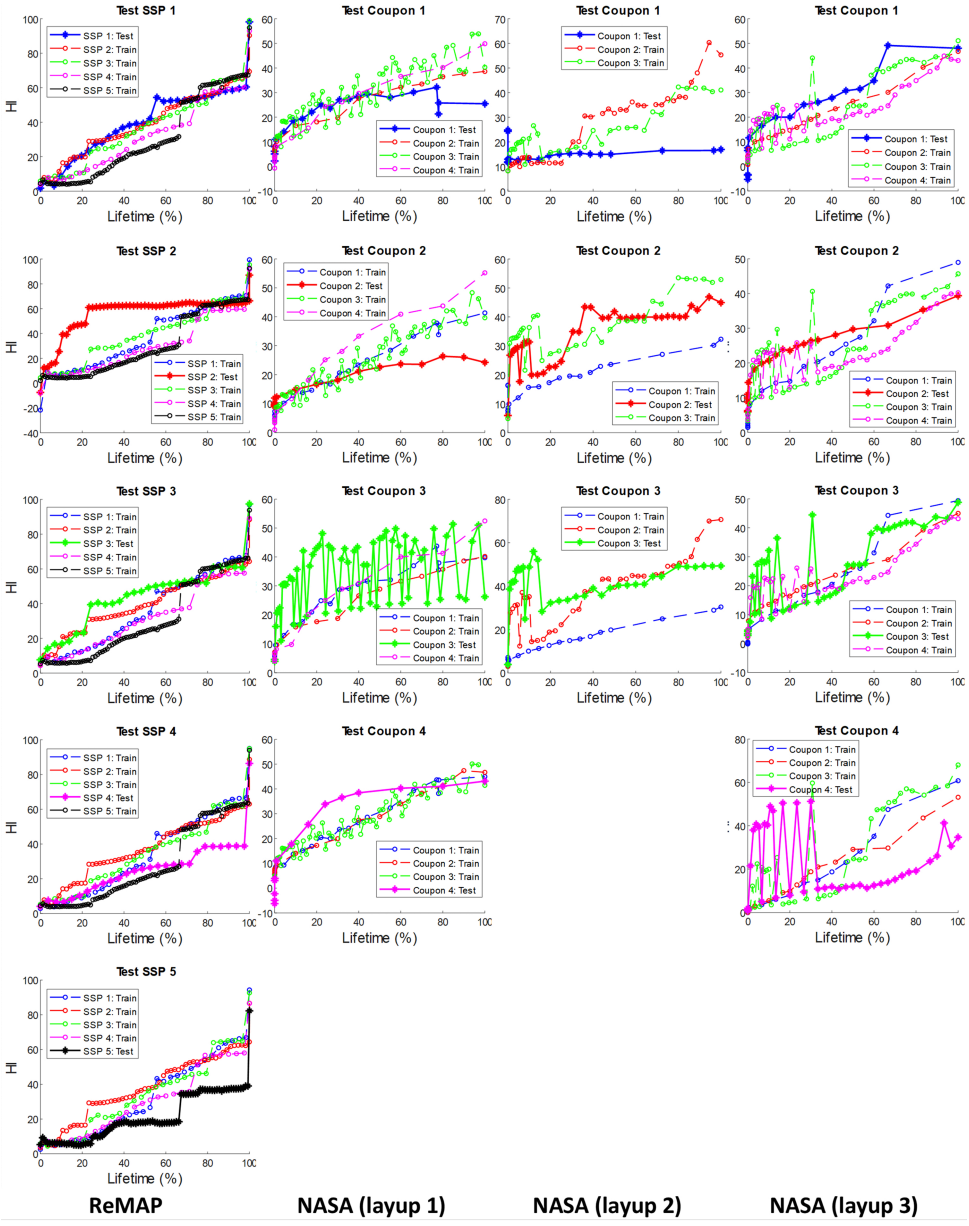


Figure 6.7: HIs obtained by the proposed framework with WAE-Fitness for different datasets, given all frequency inputs.

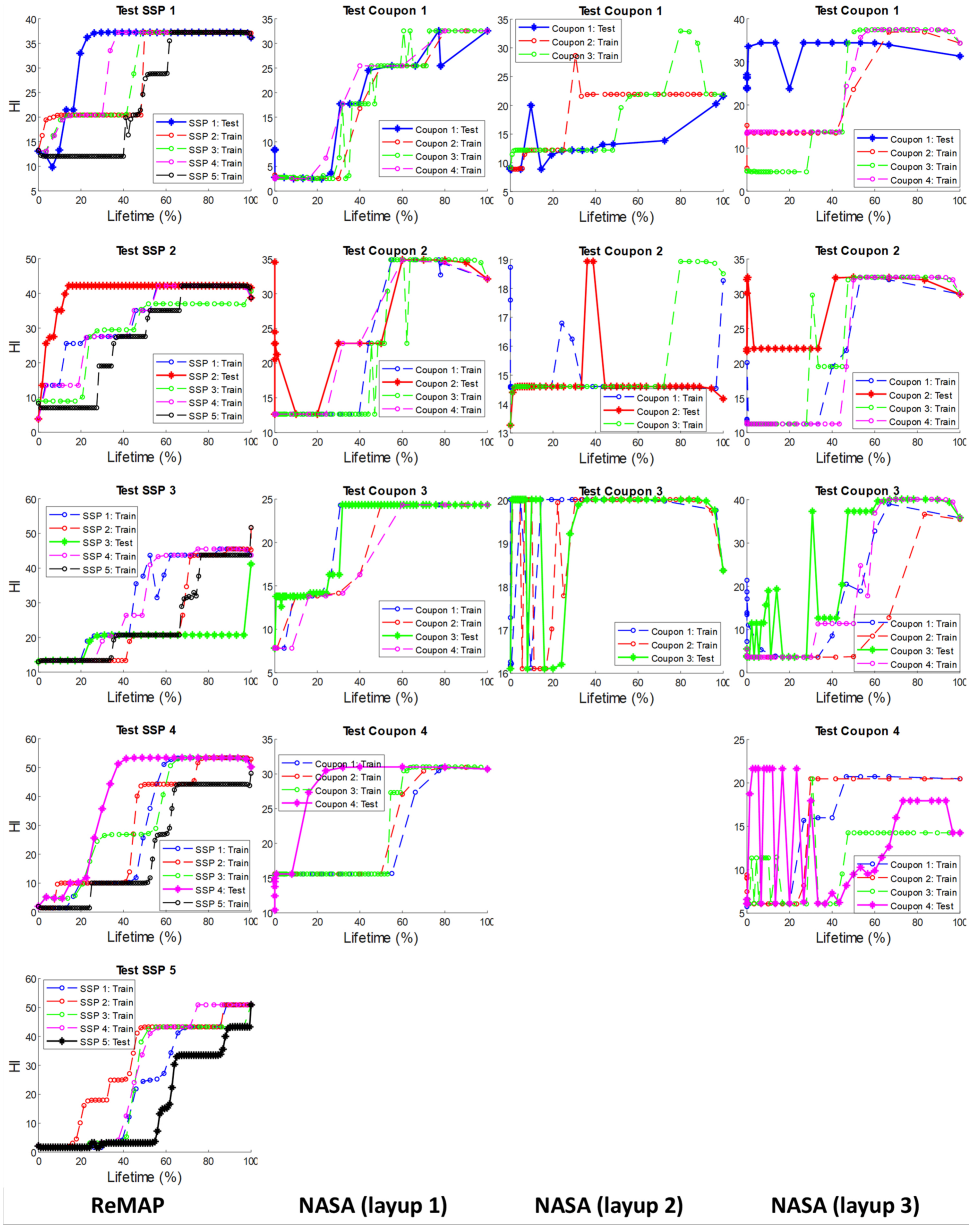


Figure 6.8: HIs obtained by the proposed framework with BiLSTM-EL for different datasets, given all frequency inputs.

6.4 Conclusions

In this chapter, an innovative approach was introduced to construct comprehensive HIs for composite structures, addressing the challenges posed by the stochastic nature of damage accumulation during operational conditions and the need for HIs independent of historical data. Leveraging the power of AI, a Hilbert transform-convolutional neural network (HT-SSCNN) was developed within the SSL paradigm. The approach exclusively utilizes current GW data, eliminating the reliance on historical information. It flexibly accommodates different GW sensor numbers, networks, and setups. The results demonstrate the effectiveness and validity of the approach. To assess various combinations and ensure robustness, rigorous evaluations were conducted, considering different datasets under various conditions.

The findings indicate that certain frequencies, such as 150 kHz for the ReMAP dataset and 250 kHz for NASA Layup 1, consistently outperformed others, resulting in more stable and reliable HIs. The use of EL techniques, specifically WAE-Fitness, led to significant improvements in HIs' performance. For the ReMAP dataset, the WAE-Fitness model, fusing all GW excitation frequencies, yielded the best fitness scores, with 93% accuracy considering all units and 89% given test units. While ReMAP experiments are more complex than the NASA ones, HIs with higher performance could be extracted from the ReMAP dataset than the NASA dataset, which can be attributed to the availability of one more training unit and a greater number of time steps for GW inspections. Additionally, ReMAP structures were monitored using a more intertwined GW sensory network, which provided a wealth of data for model training.

The produced HIs exhibit desirable properties for RUL prediction. They are monotonic, prognostable, and exhibit correlated trends, which are essential characteristics for accurate predictions in PHM. The incremental steps observed in the HIs may potentially correspond to distinct damage states, which can be used to inform state-based RUL prediction models. In conclusion, the framework offers a promising solution to the challenging task of constructing reliable and historical-independent HIs for composite structures. A high level of performance was achieved by combining AI with SP techniques, demonstrating the applicability of the method across different datasets.

References

[1] A. Saxena, K. F. Goebel, C. C. Larrosa, V. Janapati, S. Roy, and F.-K. Chang, "Accelerated aging experiments for prognostics of damage growth in composite materials," tech. rep., 2011.

[2] M. Chiachio, J. Chiachio, A. Saxena, and K. Goebel, "Documentation for the fatigue dataset in composites," *NASA Ames Res. Center, Tech. Rep*, 2013.

[3] A. Saxena, K. Goebel, C. Larrosa, and F. Chang, "Cfrp composites dataset, nasa ames prognostics data repository," *NASA Ames Research Center, Moffett Field, CA*, 2015.

[4] A. V. Oppenheim, *Discrete-time signal processing*. Pearson Education India, 1999.

7

Fusion of Acoustic Emission and Guided Waves Techniques

In this chapter, AE and GW monitoring techniques are integrated to formulate HIs for T-stiffener composite panels using the fusion of three AE frameworks with one GW framework developed in Chapters 5 and 6, respectively. The fusion approach demonstrates the potential of combining AE and GW data for SHM applications.

7.1 Introduction

In this chapter, data from two SHM techniques—acoustic emission (AE) and guided wave (GW)—are integrated to formulate HIs for T-stiffener composite panels. The frameworks developed based on AE and GW data are combined to enhance the quality of HIs by leveraging complementary information. For this purpose, three frameworks developed using AE data (as presented in Chapter 5) and one framework developed using GW data (as provided in Chapter 6) are employed.

After detailing the experimental campaigns considered for the GW-AE fusion scenario, this chapter introduces the fusion framework, which encompasses resampling and synchronization, followed by the fusion models. The results of fusing different AE frameworks and the GW framework are presented, demonstrating high performance of up to 97%, 93%, and 94% for fusion models fed by AE frameworks 1, 2, and 3, respectively, based on HIs' criteria given all units. When focusing exclusively on test units for the calculation of HIs' criteria, the fusion framework continues to exhibit strong performance, achieving up to 97%, 87%, and 89%, respectively. These results underscore the effectiveness of the fusion approach in enhancing the overall performance of SHM techniques.

7.2 Experimental Campaigns

As mentioned earlier, the ReMAP dataset includes T-stiffener composite panels monitored by AE (Chapter 5) and GW (Chapter 6). However, both monitoring systems were not available for all units, reflecting a common scenario in real-world situations. In total, 14 units underwent run-to-failure fatigue loading, with proper, available AE data for 12 units and 5 units monitored using the GW system. Among these units, only 3 were common to both AE and GW. Consequently, training GW-AE fusion models with this limited number of units is very challenging.

Reserving one unit for the testing phase of the fusion models leaves only 2 units for the training (and validation) phase. To address this constraint, as illustrated in Figure 7.1, the AE frameworks are trained on (12-1=) 11 available units with AE data (excluding the unit selected to test the fusion model), while the GW framework is trained on (5-1=) 4 available units monitored by the GW system (excluding the unit chosen for testing the fusion model, which remains the same as before). Subsequently, the trained AE and GW frameworks, along with the (3-1=) 2 available units for training that include both AE and GW data, are fed into the fusion framework.

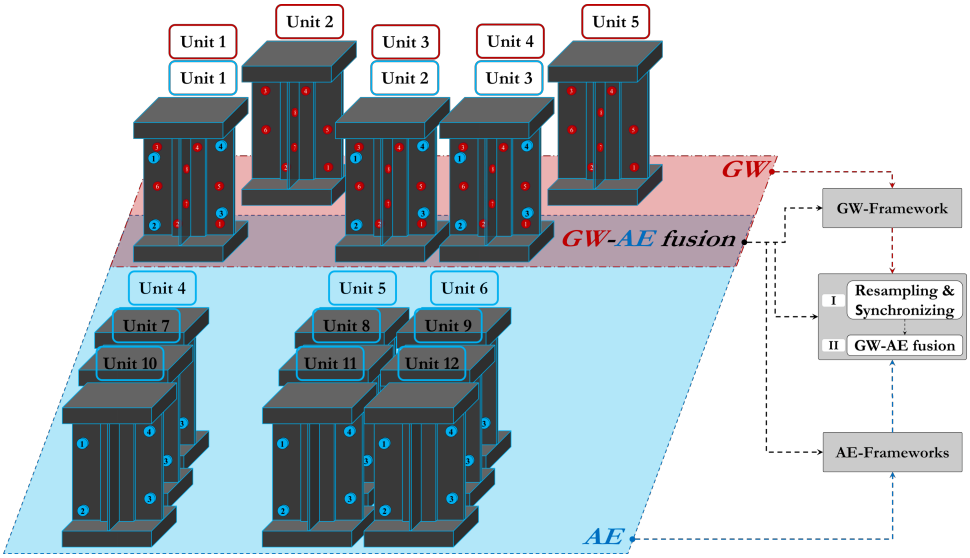


Figure 7.1: Specimens grouping based on units available for the fusion framework incorporating AE and GW data.

7.3 Fusion Framework

The fusion framework is depicted in Figure 7.2. As AE is a passive monitoring technique and GW is an active one, with different sampling frequencies and windowing rates, the HIs produced by the developed frameworks for each must undergo resampling and synchronization before being inputted into the GW-AE fusion step.

7.3.1 Resampling and synchronization

The AE data were windowed with lengths and intervals of either 500 cycles (1^{st} and 2^{nd} AE frameworks) or 1000 cycles (3^{rd} AE framework). In contrast, GW data were collected at intervals of 5000 cycles. Consequently, the number of time windows (HI quantities) for AE exceeds those for GW. To address this discrepancy, GW data are resampled and synchronized based on the time vector of AE windowed data. This process involves copying GW data from the nearest preceding neighbor, which is practical in real-world applications. The results of this resampling and synchronization step for the fusion of AE (Framework 1) and GW data across three available folds (units) are illustrated in Figure 7.3.

7.3.2 Fusion models

Various regression models, including Gaussian process regression (GPR) [1], least-squares boosting (LSBoost) [2], binary decision tree (Tree) [3], linear model using stepwise regression (LR_S) [4], robust linear regression using the bisquare weight function (LR_R) [5], SVM [6], an MLP network, and an LSTM network, are employed to integrate the HIs obtained from AE and GW. The subsequent subsections will detail the adjusted (hy)parameters for each fusion model.

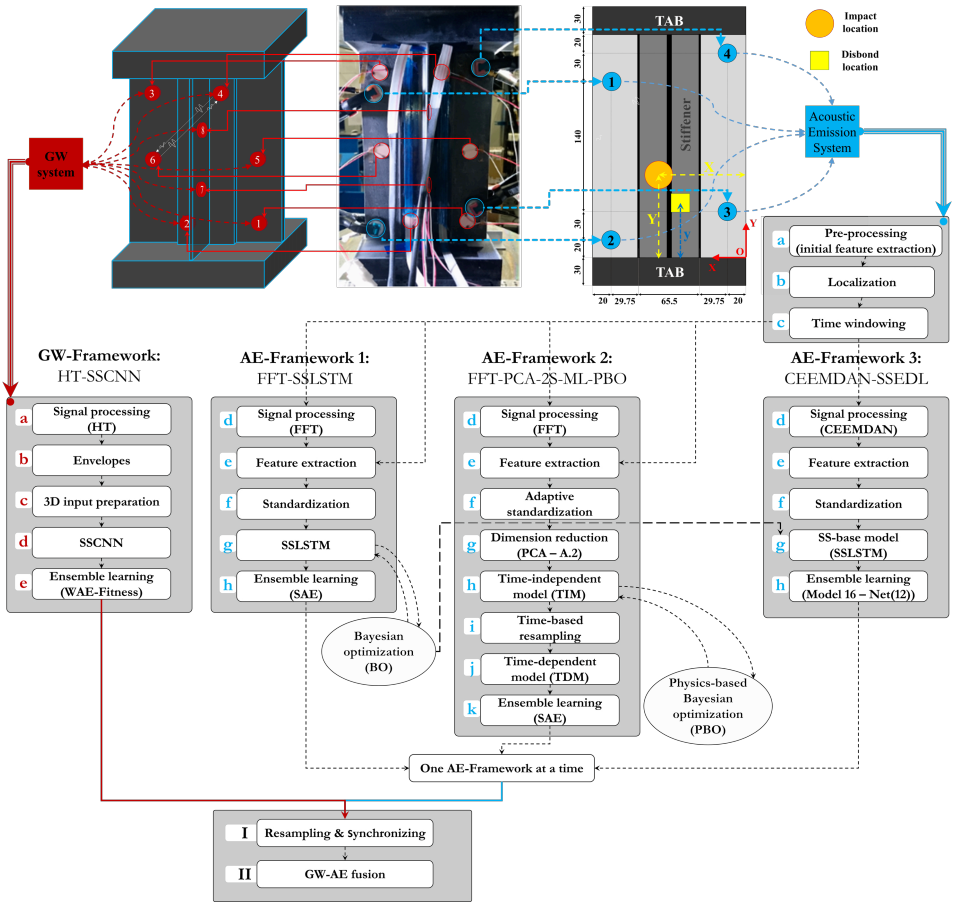


Figure 7.2: Fusion framework of AE and GW SHM monitoring systems.

GPR

A squared exponential kernel function with a constant basis function is employed for GPR. Optimization of GPR is conducted with respect to the noise standard deviation σ using BO within a 5-fold cross-validation, considering 2 units for training. The five folds are divided based on the number of data points (time steps) of the two units after shuffling. Parameters of BO include an exploration ratio of 0.5, 4 seed points, and a maximum of 30 objective evaluations. The tuned hyperparameter for noise standard deviation σ, determined through the MSE loss function, is found to be 0.6750.

LSBoost

The hyperparameters of the ensemble learning (EL) fusion model, encompassing the choice of EL method between two options—Bagging and LSBoost—as well as the number of EL cycles (search range: [10, 500]), learning rate (search range: [0.001, 1]), and the minimum leaf size of the tree as the specified learner, are tuned using BO within a 5-fold cross-validation, with consideration for 2 units during training. The folds are distributed

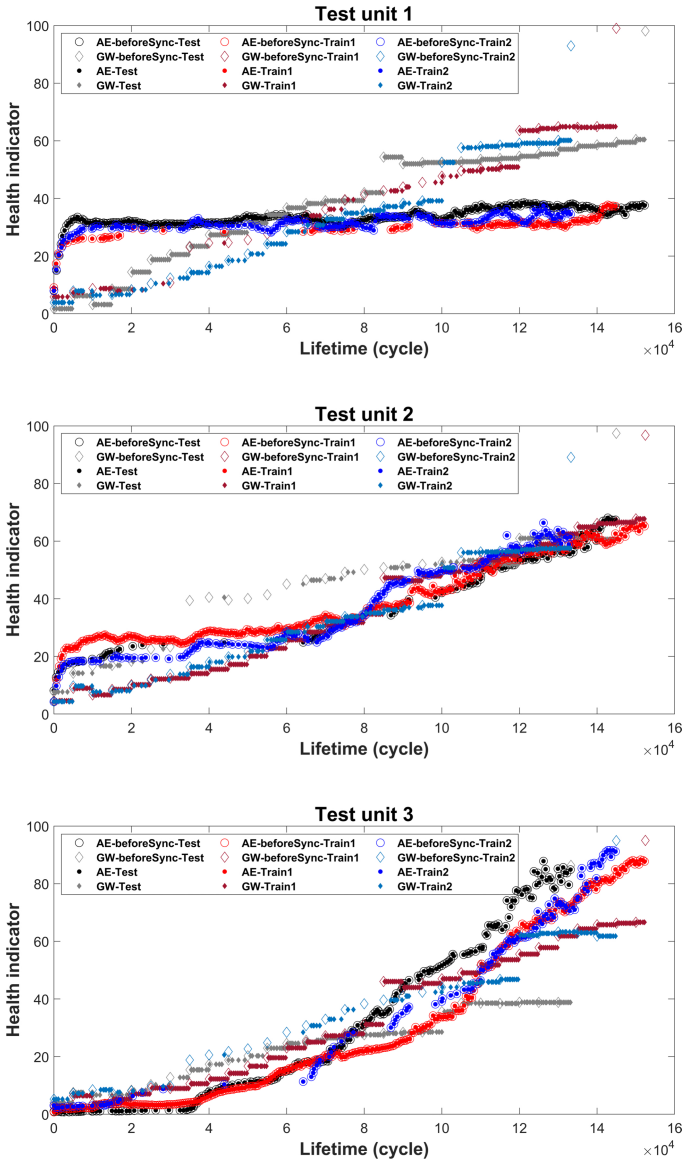


Figure 7.3: HIs derived from AE (Framework 1) and GW data, both before and after undergoing the resampling and synchronization process, for three folds (units) designated for GW-AE fusion.

based on the number of data points (time steps) of the two units after shuffling. BO parameters include an exploration ratio of 0.5, 4 seed points, and a maximum of 30 objective evaluations. The optimized hyperparameters for the EL method were determined through the MSE loss function, resulting in LSBoost, 457 cycles, a 0.1383 learning rate, and a minimum leaf size of 4.

Tree

The minimum leaf size of the binary decision tree, considered its hyperparameter with a search range of [1, 149], is tuned using BO within a 5-fold cross-validation, following the same procedure as the other fusion models. The optimized minimum leaf size is found to be one, and the resulting binary decision tree has 99 nodes with node sizes ranging from 1 to 298.

LR_S

Stepwise linear regression (LR_S) utilizes the p-value from an F-statistic to assess models by adding or removing potential terms at each step. When a term is absent in the model, the null hypothesis assumes that the term would have a zero coefficient if included. If there is enough evidence to reject this null hypothesis, the function incorporates the term into the model. Conversely, when a term is already present in the model, the null hypothesis assumes a zero coefficient for that term. If there is inadequate evidence to reject this null hypothesis, the function removes the term. The p-value, derived from an F-test assessing the change in sum of squared error resulting from adding or removing the term, is used to train LR_S. The criteria thresholds for adding or removing a term are set at 0.05 and 0.1, respectively. With a maximum of 1000 steps, the LR_S fusion model undergoes training within a 5-fold cross-validation.

LR_R

A robust linear regression model, denoted as LR_R, is fitted using a bisquare weight function. In this study, the model is trained within a 5-fold cross-validation to fuse AE-HIs with GW-HIs, aligning with the ideal simulated HIs.

SVM

An SVM regression model with a linear kernel, ϵ set to 3.727, and utilizing sequential minimal optimization, is trained within a 5-fold cross-validation to regress the inputs to the ideal simulated HIs.

MLP

An MLP with three hidden layers, comprising 5, 3, and 1 neurons, is trained using Levenberg-Marquardt backpropagation. The activation function log-sigmoid is employed for the hidden layers, while the linear activation function is used for the output layer. With a loss function of MSE and allocating 30% of the total number of data points from the two units considered for training as validation, the MLP model is trained to fuse HIs.

LSTM

An LSTM model, as depicted in Figure 7.4, is specifically designed to integrate GW-HIs with AE-HIs. Through iterative trials and adjustments, the configuration and hyperparameters have been fine-tuned. The model utilizes the Adam optimizer with an initial learning rate of 0.1. To mitigate overfitting, a regularization term, known as weight decay or L2 regularization, is introduced with a set value of 0.001. Despite a maximum training epoch of 2000, the network's output is determined based on the best validation loss, with a validation check frequency set at 30 iterations and a validation check patience set to 50. Data shuffling occurs at each epoch, with a mini-batch size of 20 ensuring robust training and fusion performance within the LSTM architecture.

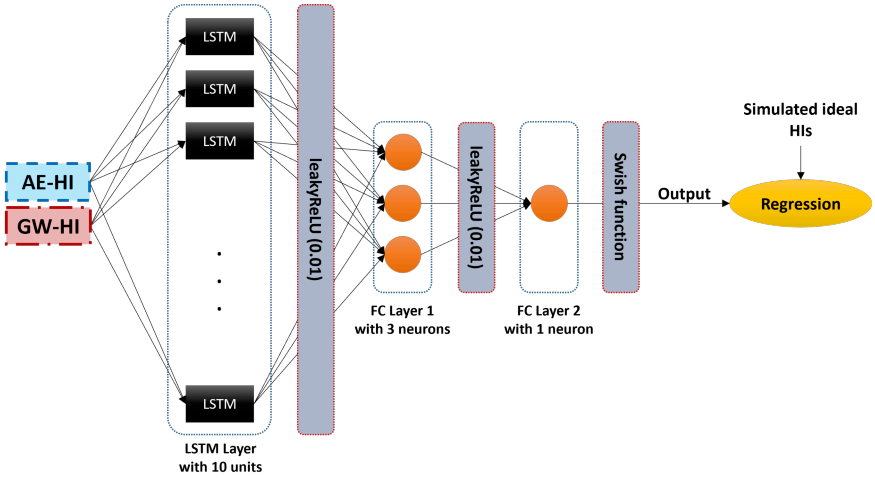


Figure 7.4: LSTM model designed for GW-AE fusion.

7.4 Results

To comprehensively assess various combinations and validate the effectiveness and stability of the proposed framework, all potential dataset folds were examined. In each fold, one unit was designated as the test set, while the remaining two composite panels were utilized for training (and validation). The HIs constructed through the fusion of the 1^{st}, 2^{nd}, and 3^{rd} AE frameworks with the GW framework are illustrated in Figures 7.5 to 7.7, respectively.

Regarding the fusion of the 1^{st} AE framework with the GW framework, it is notable that except for Unit 1, where GW produces superior HIs, AE data yields higher fitness scores for the other two units. Specifically, when considering fitness scores exclusively for test units, the LSTM model outperforms others. However, in the case of fitness given all units, LR_R, MLP, and LSTM demonstrate the highest fitness scores for Units 1, 2, and 3, respectively.

Concerning the fusion of the 2^{nd} AE framework with the GW framework, noteworthy observations include the superior HIs produced by GW on average compared to AE data. In terms of fitness scores across all units, MLP achieves a higher score for fold 1 (Test SSP 1). However, it is evident that the HIs for the training units are overfitted, and the HI for test unit 1 deviates. This situation poses a challenge for prediction. Conversely, based on fitness scores given the test unit, LR_R attains a higher score for fold 1. Notably, the HIs are not overfitted, and all represent a distinctive jump with a consistent increasing pattern afterward. This pattern may potentially signify different damage states during the fatigue loading of composite structures. This underscores the importance of using refined criteria specifically designed for test units. While LR_R provides better scores for the test unit in folds 1 and 2, it does not for fold 3. Consequently, LSTM is preferred, as its scores for folds 1 and 2 are not significantly lower, and it outperforms others for fold 3. This choice reflects the adaptability and effectiveness of LSTM in capturing the varying patterns across different folds.

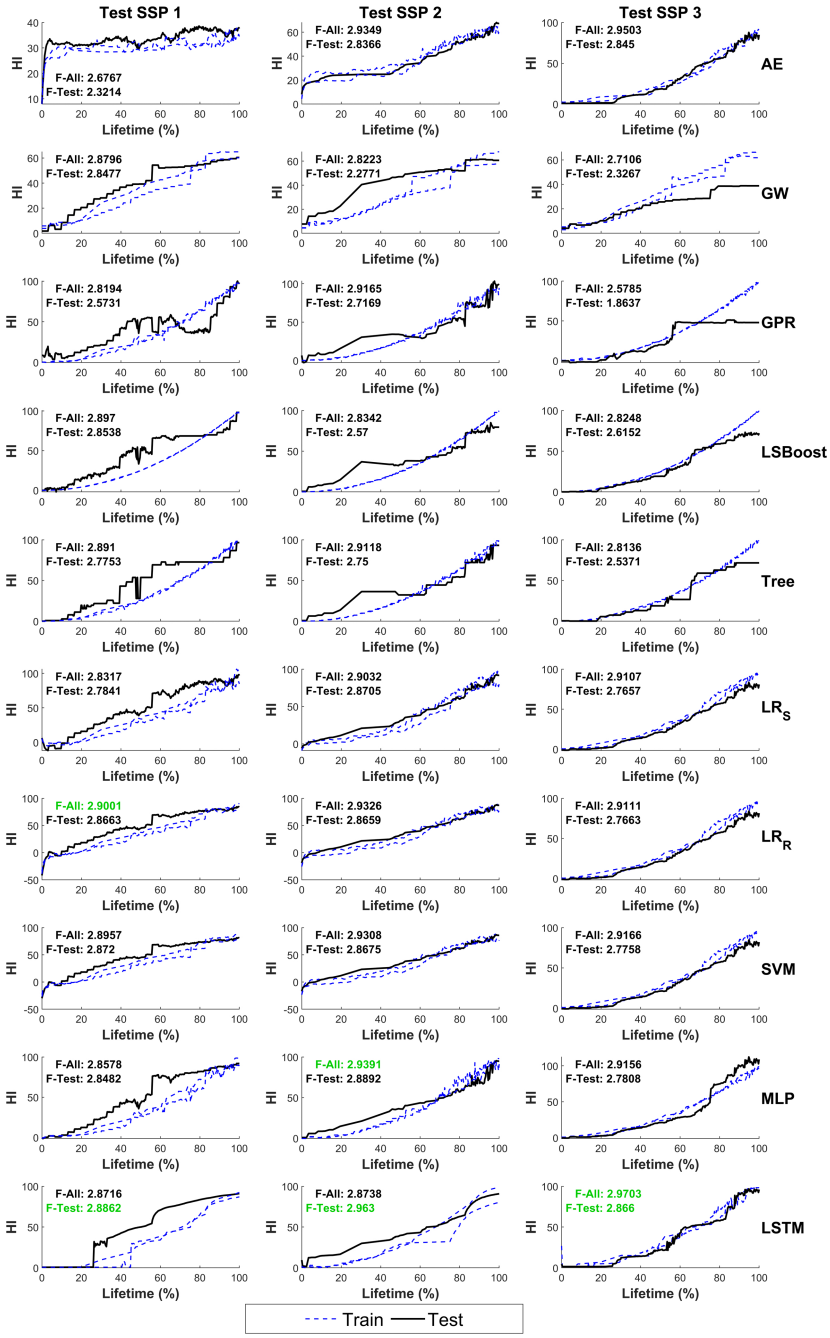


Figure 7.5: HIs obtained through the fusion framework combining the 1^{st} AE framework and the GW framework. Columns correspond to specific folds labeled at the top, while rows represent the same SHM inputs (AE or GW) or fusion models, labeled on the right. Green-colored values highlight the maximum fitness score obtained given all (F-All) or test (F-Test) units for the fold indicated at the top.

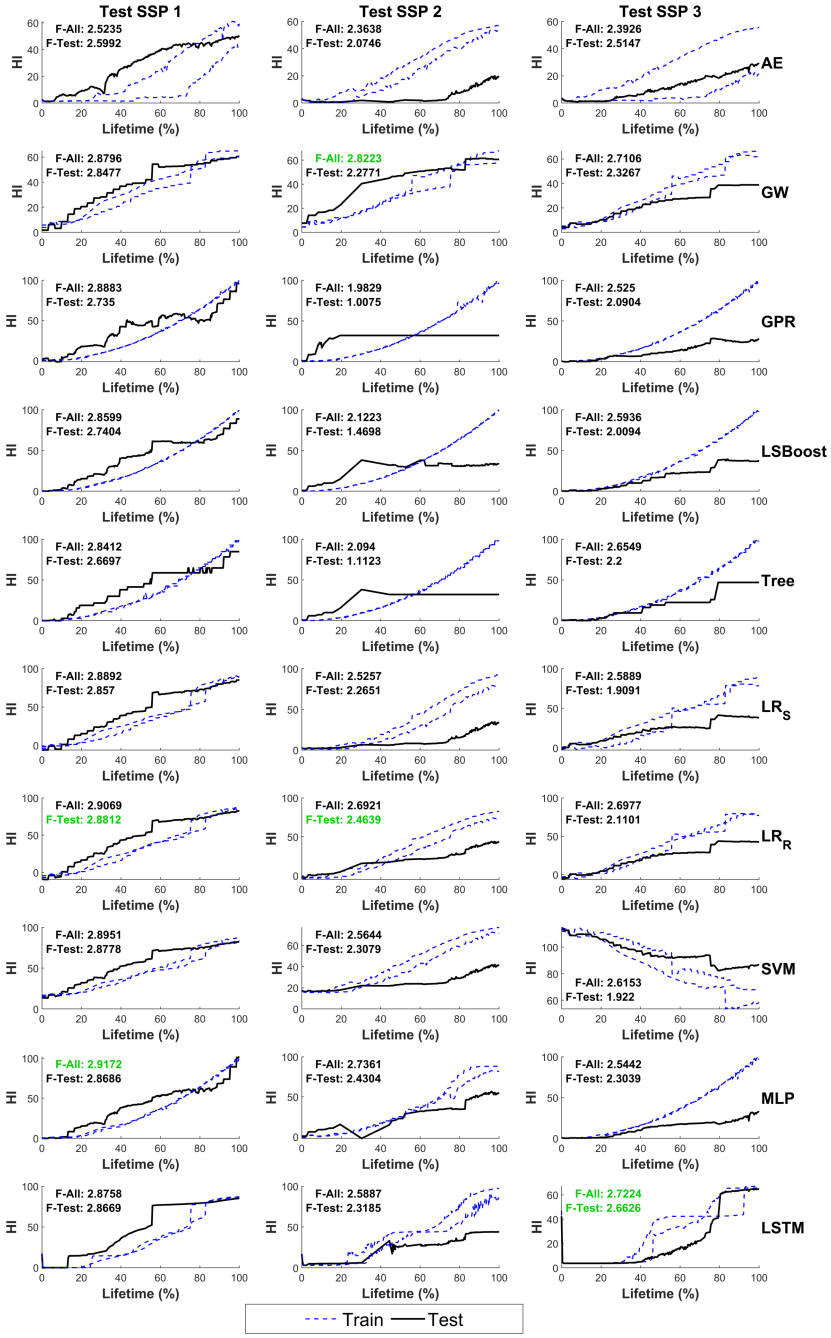


Figure 7.6: HIs obtained through the fusion framework combining the 2^{nd} AE framework and the GW framework. Columns correspond to specific folds labeled at the top, while rows represent the same SHM inputs (AE or GW) or fusion models, labeled on the right. Green-colored values highlight the maximum fitness score obtained given all (F-All) or test (F-Test) units for the fold indicated at the top.

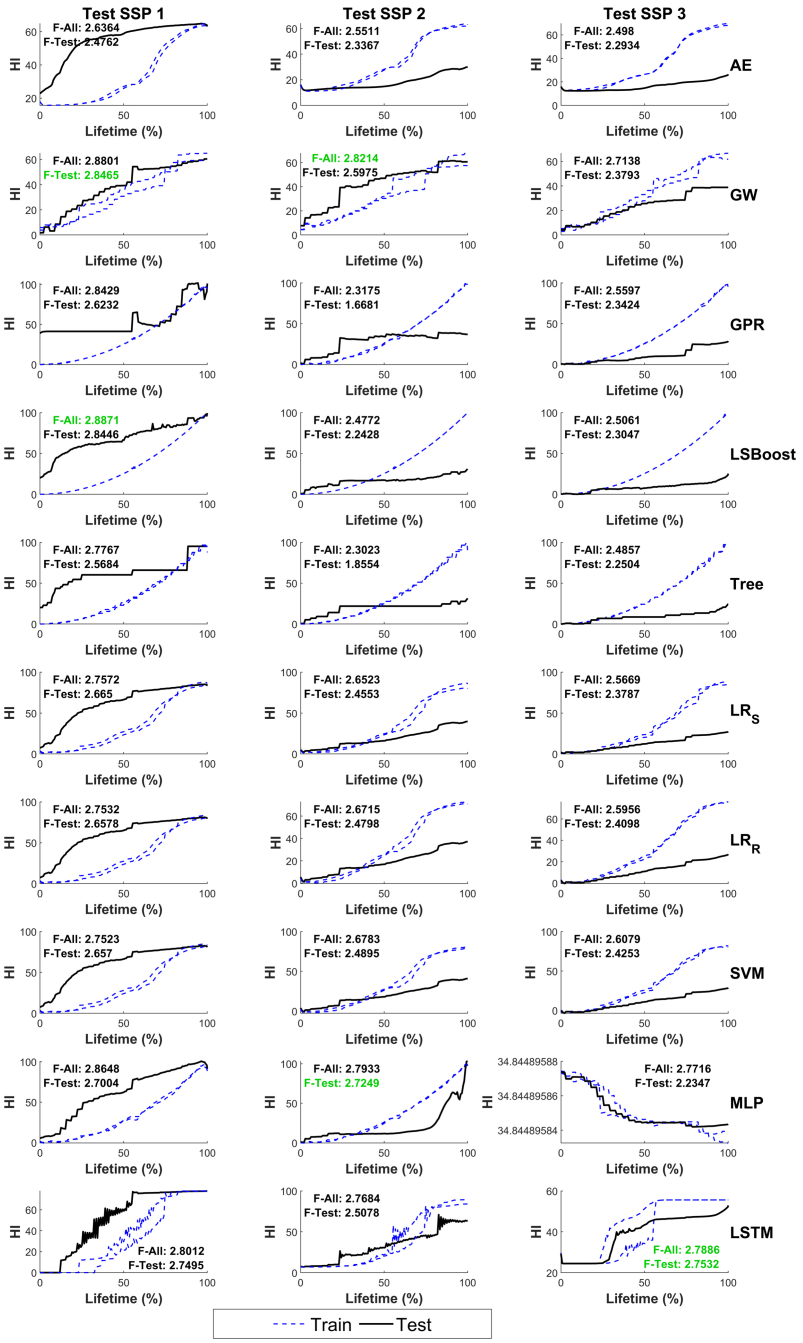


Figure 7.7: HIs obtained through the fusion framework combining the 3^{rd} AE framework and the GW framework. Columns correspond to specific folds labeled at the top, while rows represent the same SHM inputs (AE or GW) or fusion models, labeled on the right. Green-colored values highlight the maximum fitness score obtained given all (F-All) or test (F-Test) units for the fold indicated at the top.

Concerning the fusion of the 3^{rd} AE framework with the GW framework, several observations emerge. For the first fold, the fusion of AE and GW does not outperform the GW-HI, but the LSTM fusion model stands out as the best for fold 3. In fold 2, the high F-Test score of MLP is attributed to the high Pr compared to GW. The HIs of fold 1 obtained from LSTM are particularly interesting. Initially, the HIs do not exhibit any increase, representing a healthy state. Subsequently, after a distinctive jump, they demonstrate a non-linear increasing pattern with fluctuations, indicative of a middle damage state. The final damage state includes a consistent, slightly increasing pattern leading up to the eventual failure. Notably, the duration of the middle damage state lasts around 40-50% of the EoL.

A detailed breakdown of the HIs' criteria averaged over three folds is presented in Table 7.1. According to the Fitness-Test metric, the LSTM fusion model emerges as the most effective across all three AE frameworks. However, it is worth noting that the scores for the 1^{st} AE framework are higher than the other two due to the use of the test unit for validation, as explained in Chapter 5. This limitation hinders the generalizability of the developed framework.

The Mo-Test scores for GW-based HI differ for Framework 3 compared to the other two, indicating a discrepancy that should be addressed. This inconsistency is attributed to the resampling process, highlighting the need for improvement in this metric in future work because the criteria should not be affected by the number of time steps.

In summary, GW-based HIs outperform AE-based HIs, even though they were trained on fewer units (11 units for AE and 4 for GW). Nevertheless, AE, as a passive SHM technique, offers higher temporal resolution.

The overall results indicate that the GW-AE fusion step enhances Fitness scores compared to using only one SHM technique. However, considering that this improvement, compared to the best single SHM input, is less than 10% (9%, 6%, and 2% for AE frameworks 1, 2, and 3, respectively), the justifiability of this improvement must be assessed in the context of the cost of monitoring systems and computational complexity. It prompts consideration of whether using only the GW technique, but with a more intricate network of sensors and a greater number of time steps for inspection conditions, might compensate for this improvement. Thus, these findings raise new questions in the field.

Table 7.1: Distribution of HIs' criteria values across three folds for the fusion framework fed by the GW model and AE models (three frameworks) for the ReMAP dataset. The values are computed from Eqs. 3.1 - 3.4, encompassing all units denoted as '-All,' and Eqs. 3.5 - 3.8, focusing solely on test units labeled as '-Test.'

AE model	Criteria	Inputs (after resampling)		Fusion models							
		AE	GW	GPR	LSBoost	Tree	LR_S	LR_R	SVM	MLP	LSTM
Framework 1	Mo-All	0.99 (±0.02)	1 (±0)	0.99 (±0.01)	1 (±0)	1 (±0)	1 (±0)	1 (±0)	1 (±0)	1 (±0)	1 (±0)
	Mo-Test	0.81 (±0.18)	0.7 (±0.28)	0.65 (±0.2)	0.88 (±0.07)	0.85 (±0.03)	0.9 (±0.02)	0.92 (±0.01)	0.92 (±0.02)	0.93 (±0.02)	0.97 (±0.05)
	Pr-All	0.95 (±0.01)	0.87 (±0.11)	0.89 (±0.17)	0.9 (±0.07)	0.93 (±0.08)	0.91 (±0.01)	0.94 (±0.02)	0.94 (±0.02)	0.95 (±0.02)	0.95 (±0.05)
	Pr-Test	0.94 (±0.01)	0.85 (±0.2)	0.84 (±0.26)	0.85 (±0.13)	0.89 (±0.13)	0.94 (±0.06)	0.94 (±0.07)	0.94 (±0.06)	0.96 (±0.06)	0.98 (±0.01)
	Tr	0.92 (±0.12)	0.94 (±0.03)	0.9 (±0.05)	0.95 (±0.04)	0.94 (±0.04)	0.97 (±0.03)	0.97 (±0.03)	0.97 (±0.03)	0.96 (±0.04)	0.96 (±0.05)
	Fitness-All	2.85 (±0.15)	2.8 (±0.09)	2.77 (±0.17)	2.85 (±0.04)	2.87 (±0.05)	2.88 (±0.04)	2.91 (±0.02)	2.91 (±0.02)	2.9 (±0.04)	**2.91 (±0.06)**
	Fitness-Test	2.67 (±0.3)	2.48 (±0.32)	2.38 (±0.46)	2.68 (±0.15)	2.69 (±0.13)	2.81 (±0.06)	2.83 (±0.06)	2.84 (±0.05)	2.84 (±0.05)	**2.91 (±0.05)**
Framework 2	Mo-All	1 (±0)	1 (±0)	0.98 (±0.02)	0.99 (±0.01)	0.98 (±0.02)	1 (±0.01)	1 (±0.01)	0.99 (±0.01)	1 (±0)	1 (±0)
	Mo-Test	0.93 (±0.04)	0.7 (±0.28)	0.59 (±0.37)	0.67 (±0.22)	0.58 (±0.37)	0.79 (±0.28)	0.81 (±0.24)	0.78 (±0.3)	0.91 (±0.08)	0.95 (±0.05)
	Pr-All	0.67 (±0.16)	0.87 (±0.11)	0.72 (±0.23)	0.73 (±0.18)	0.74 (±0.16)	0.76 (±0.19)	0.82 (±0.14)	0.78 (±0.17)	0.79 (±0.19)	0.85 (±0.2)
	Pr-Test	0.71 (±0.28)	0.85 (±0.2)	0.59 (±0.33)	0.59 (±0.27)	0.61 (±0.23)	0.65 (±0.27)	0.73 (±0.21)	0.68 (±0.25)	0.68 (±0.27)	0.79 (±0.29)
	Tr	0.76 (±0.08)	0.94 (±0.03)	0.76 (±0.3)	0.81 (±0.24)	0.81 (±0.24)	0.91 (±0.01)	0.95 (±0.02)	0.92 (±0.01)	0.94 (±0)	0.88 (±0.1)
	Fitness-All	2.43 (±0.09)	**2.8 (±0.09)**	2.47 (±0.46)	2.53 (±0.37)	2.53 (±0.39)	2.67 (±0.19)	2.77 (±0.12)	2.69 (±0.18)	2.73 (±0.19)	2.73 (±0.14)
	Fitness-Test	2.4 (±0.28)	2.48 (±0.32)	1.94 (±0.87)	2.07 (±0.64)	1.99 (±0.8)	2.34 (±0.48)	2.49 (±0.39)	2.37 (±0.48)	2.53 (±0.3)	**2.62 (±0.28)**
Framework 3	Mo-All	0.99 (±0)	1 (±0)	0.99 (±0.01)	1 (±0)	0.99 (±0.01)	1 (±0)	1 (±0)	1 (±0)	0.99 (±0.01)	0.99 (±0)
	Mo-Test	0.93 (±0.09)	0.82 (±0.12)	0.75 (±0.23)	0.97 (±0.02)	0.81 (±0.13)	0.95 (±0.06)	0.95 (±0.06)	0.95 (±0.06)	0.76 (±0.24)	0.92 (±0.08)
	Pr-All	0.7 (±0.25)	0.87 (±0.11)	0.73 (±0.2)	0.72 (±0.24)	0.71 (±0.21)	0.75 (±0.2)	0.76 (±0.19)	0.77 (±0.18)	0.93 (±0.06)	0.92 (±0.09)
	Pr-Test	0.58 (±0.36)	0.85 (±0.2)	0.6 (±0.3)	0.59 (±0.35)	0.59 (±0.33)	0.63 (±0.31)	0.65 (±0.3)	0.66 (±0.29)	0.9 (±0.08)	0.88 (±0.14)
	Tr	0.86 (±0.18)	0.94 (±0.03)	0.85 (±0.14)	0.91 (±0.03)	0.83 (±0.11)	0.91 (±0.11)	0.91 (±0.12)	0.91 (±0.12)	0.89 (±0.04)	0.87 (±0.07)
	Fitness-All	2.56 (±0.07)	2.81 (±0.08)	2.57 (±0.26)	2.62 (±0.23)	2.52 (±0.24)	2.66 (±0.1)	2.67 (±0.08)	2.68 (±0.07)	**2.81 (±0.05)**	2.79 (±0.02)
	Fitness-Test	2.37 (±0.1)	2.61 (±0.23)	2.21 (±0.49)	2.46 (±0.33)	2.22 (±0.36)	2.5 (±0.15)	2.52 (±0.13)	2.52 (±0.12)	2.55 (±0.28)	**2.67 (±0.14)**

7.5 Conclusions

In summary, the fusion of AE and GW data using various regression models, particularly the LSTM model, demonstrates promise for T-stiffener composite panel health monitoring. Synchronization and resampling of GW data based on AE windowed data were required for fusion. Despite marginal improvements in Fitness scores, questions arise about the cost-effectiveness of the fusion approach compared to using only the GW technique with a more intricate sensor network. The study provides valuable insights into the challenges and potential trade-offs associated with AE-GW fusion in SHM.

References

[1] C. E. Rasmussen, C. K. Williams, *et al.*, *Gaussian processes for machine learning*, vol. 1. Springer, 2006.

[2] T. Hastie, R. Tibshirani, J. H. Friedman, and J. H. Friedman, *The elements of statistical learning: data mining, inference, and prediction*, vol. 2. Springer, 2009.

[3] L. Breiman, *Classification and regression trees*. Routledge, 2017.

[4] N. R. Draper and H. Smith, *Applied regression analysis*, vol. 326. John Wiley & Sons, 1998.

[5] P. W. Holland and R. E. Welsch, "Robust regression using iteratively reweighted least-squares," *Communications in Statistics-theory and Methods*, vol. 6, no. 9, pp. 813–827, 1977.

[6] V. Vapnik, *The nature of statistical learning theory*. Springer science & business media, 1999.

8

Conclusions and Recommendations

8.1 Conclusions

This research embarked on a comprehensive exploration of HI design for SHM and prognostics, focusing primarily on composite structures. The investigation spanned various methodologies, incorporating AI, SP, and innovative fusion approaches. By focusing on key evaluation criteria such as Mo, Pr, and Tr and identifying critical gaps like the need for nonlinear HIs, generalizability, and historical-independent HIs, the groundwork was laid for targeted investigations.

In response to these challenges, novel approaches were introduced to enhance HIs. The emphasis on test units for modifying evaluation criteria and the introduction of a SSL paradigm derived from nonlinear HIs, aligned with the physics of damage propagation, marked notable methodological contributions. Additionally, the integration of a modified multiplicative neuron with sparsity control for discretizing weights aimed at designing interpretable HIs with concise equations, particularly beneficial for turbofan engines.

The main conclusion of the thesis can be listed as follows:

- Three frameworks integrating AI and SP were developed to formulate HIs for T-stiffener composite panels using AE data. The frameworks demonstrated superior performance, 93%, 90%, and 91% based on HIs' criteria given all units, respectively, outperforming existing literature models. Emphasis on improved feature extraction by advanced SP methods, optimizing the ANN architectures and hyperparameters by BO and PBO, implementing the HIs' criteria into the learning and optimizing process, separating the tasks of temporal and spatial information extraction during the learning process, and ensemble learning showcased advancements in HI design.

- The implementation of a HT-SSCNN fed by GW data followed by ensemble learning demonstrated high performance and highlighted the potential of data-driven and AI approaches in health monitoring applications for composite structures. The methodology is validated through investigations on T-single stiffener CFRP panels under compression-fatigue and dogbone CFRP specimens under tension-fatigue loadings, showing high performance of up to 93% and 81% based on HIs' criteria given all units, respectively. The proposed GW approach exclusively utilizes current GW data, eliminating the reliance on historical information. It flexibly accommodates different GW sensor numbers, networks, and setups. The incremental steps observed in some HIs may potentially correspond to distinct damage states, representing another aspect of interpretability that can be used to inform state-based RUL prediction models.

- The fusion framework of AE and GW techniques demonstrated the potential of combining these techniques for SHM applications. The results of fusing different developed AE frameworks and the GW framework are presented, demonstrating high performance of up to 97%, 93%, and 94% for fusion models fed by AE frameworks 1, 2, and 3, respectively, based on HIs' criteria given all units. When focusing exclusively on test units for the calculation of HIs' criteria, the fusion framework continues to exhibit strong performance, achieving up to 97%, 87%, and 89%, respectively. These results underscore the effectiveness of the fusion approach in enhancing the

overall performance of SHM techniques. However, questions arose about the cost-effectiveness of the fusion approach compared to using only the GW technique.

- Lastly, INN was developed for designing HIs for commercial turbofan engines. The introduction of modified multiplicative layers alongside discretizing weights by sparsity control provided a breakthrough in creating concise equations. The performance of the INN surpassed that of alternative methods while maintaining interpretability in HI design.

8.2 Recommendations

While the research has made significant strides in addressing gaps in HI design and application, several areas for future exploration and improvement are identified:

Cost-effectiveness and computational complexity analysis

Given the potential trade-offs of using fusion approaches, a detailed cost-effectiveness analysis is recommended. Comparing the performance gains with the cost and complexity of monitoring systems will provide valuable insights for decision-making in practical applications. Also, while the fusion of AE and GW techniques showcased improvements in performance, the computational complexity of fusion approaches warrants further investigation. Evaluating the trade-offs between enhanced performance and computational efficiency would provide valuable insights for practical implementation. Additionally, the comparison should be fair, meaning that the same (enough) number of units without missing information should be available to train the AE-HI and GW-HI construction frameworks.

Adaptive-active fusion scenario

An efficient fusion scenario, considering a cost-effective combination of SHM techniques, could be an adaptive-active monitoring system. This system activates the active SHM technique (e.g., guided wave) when the passive SHM technique (e.g., acoustic emission) detects a noticeable variation due to health degradation. This adaptive-active fusion scenario leverages temporal information recorded from the passive SHM technique efficiently, minimizing the number of active SHM technique assessments and reducing operational shutdown times.

Criteria improvement

As observed in Chapter 7, the resampling process affected Mo, highlighting the need for improvement in this metric in future work to ensure that criteria are not influenced by the number of time steps.

Optimizing the weight of Tr that contributes to Fitness would enhance the flexibility of learning algorithms. The rationale behind this lies in the author's belief that Tr directly correlates with uncertainty and may not always achieve its maximum value in a comprehensive health indicator.

Additionally, criteria should be interconnected in a conditional manner, ensuring that HIs not only meet the requirements of Pr and Tr but also incorporate health degradation, thereby enhancing their overall utility. For instance, if HIs exhibit purely horizontal

patterns without any increase or decrease, they may meet the criteria of Pr and Tr with maximum scores. However, such HIs might lack practical significance since they do not indicate any health degradation. In this context, Mo should be conditional to confirm Pr and Tr. The author believes that further enhancement of the criteria is possible.

Integration of uncertainty

Future work should explore methods to capture and integrate uncertainty within models towards better decision-making. Differentiating noise or uncertainty components and incorporating them into the frameworks can enhance stability and reliability, contributing to the development of more robust HIs.

8.3 Final Remarks

In conclusion, this thesis has advanced the field of prognostics and health management, with a specific focus on health indicator design, particularly within the domain of composite structures. The innovative methodologies, which span artificial intelligence, signal processing, and fusion techniques, have consistently demonstrated superior performance. This research has also initiated discussions and taken important steps towards achieving interpretability in health indicator design.

The methodologies, frameworks, and insights presented in this thesis lay a solid foundation for further research, contributing to the ongoing innovation in health monitoring applications. The recommendations provided offer valuable guidance for future research directions, with the aim of refining and extending the proposed frameworks for broader applications in engineering systems.

The journey undertaken, from exploring critical gaps in the literature to proposing advanced methodologies, has played a pivotal role in the continuous evolution of prognostics and health management. As technology and methodologies continue to evolve, the exploration of reliable and interpretable health indicators will remain crucial to ensuring the safety and efficiency of complex engineering systems, such as composite structures.

A

Appendices

A

A.1 Interpretable HIs for Commercial Turbofan Engines

This section aims to develop an interpretable ANN with the capability for automatic feature selection and fusion targeting optimal HIs with considerably lower complexity. As explained in Chapter 3 (Section 3.5), this architecture is made up of additive and new modified multiplicative layers that combine features to better represent the system's physical characteristics. To extract a compact HI equation—making the neural network mathematically interpretable—the number of parameters is further reduced by discretizing the weights via a ternary set. This weight discretization simplifies the extracted equation while softly controlling the number of weights that should be overlooked.

The proposed design is tested using the NASA Ames Prognostics Data Repository dataset for turbofan engine degradation simulation, which is widely used in the PHM field [1]. The findings will be discussed in comparison with PCA, KPCA, and two-stage genetic programming (GP) outputs. This methodology makes several key contributions, including:

1. Introducing a new type of neuron that operates multiplicatively, in addition to the commonly used additive neurons.
2. Building a network that combines both additive and multiplicative neurons to generate accurate and robust hybrid HI.
3. Utilizing the benefits of both multiplicative neurons and sparsity control by implementing discretized (ternary) weights, resulting in concise and efficient equations.
4. Developing HI models with concise and easy-to-understand equations, while ensuring they meet the evaluation criteria of Mo, Pr, and Tr.

In this section, a concise overview of pre-processing, de-noising, and data division into training and test sets is presented in Section A.1.1. Subsequently, the interpretable HI construction framework, which incorporates newly modified multiplicative layers alongside additive ones and utilizes discretized weights, is outlined. Finally, the criteria for HIs will be presented and discussed.

A.1.1 Dataset

The present section focuses on the NASA Ames Prognostics Data Repository dataset for commercial turbofan engines (CMAPSS) [1]. This dataset is generated using the C-MAPSS tool, which models various engine fleet deterioration scenarios—from a baseline condition to the point of final failure in the training data and a time period prior to the EoL in the test data. Two sets of data are investigated in this section: first, engines degrading with one failure mode (FD001); and second, engines degrading with two failure modes (FD003). Each engine's ID and deterioration time steps are given in the first and second columns, respectively. The next three columns provide the engine's operational characteristics, and the final 21 columns list the signals from 21 sensors. Both subsets FD001 and FD003 consist of 200 turbofan engine units each, with 100 designated for model training and the remaining 100 for testing RUL prediction models. However, the 100 units allocated for RUL model testing lack sensory data up to the EoL. Consequently, these units cannot be utilized for evaluating the HI construction model using the HIs' criteria (Mo, Tr,

and Pr). As a result, a test fraction equivalent to 20% of the first 100 turbofan engines with complete input sensory data until EoL in each subset is reserved for testing.

Data processing can suffer from signals that are constant throughout all measurement points. As a result, data that have identical upper and lower boundaries is first found and removed. Accordingly, out of the 21 sensors, the 1^{st}, 5^{th}, 10^{th}, 16^{th}, 18^{th}, and 19^{th} are removed, leaving 15 in place for the subset FD001. The same sensors are removed from subset FD003 with the exception of sensor 10, leaving 16 in place. As a result, the remaining sensors are denoted in the following as 1 through 16, among which the 16^{th} sensor refers to the different sensor used in subset FD003 (i.e., sensor 10), and the other sensors have the same index. Data have been standardized using a zero-mean normalization technique that used only the training samples' mean value and standard deviation. Additionally, to improve the quality of the resulting features and HI, the signals can be de-noised. In this case, a polynomial function of order four is used to perform a regression. The resulting de-noised signals (features) can then be chosen as HIs or retrieved (feature extraction) and combined (feature fusion) to create an appropriate HI.

A.1.2 Building Interpretable ANN (INN)

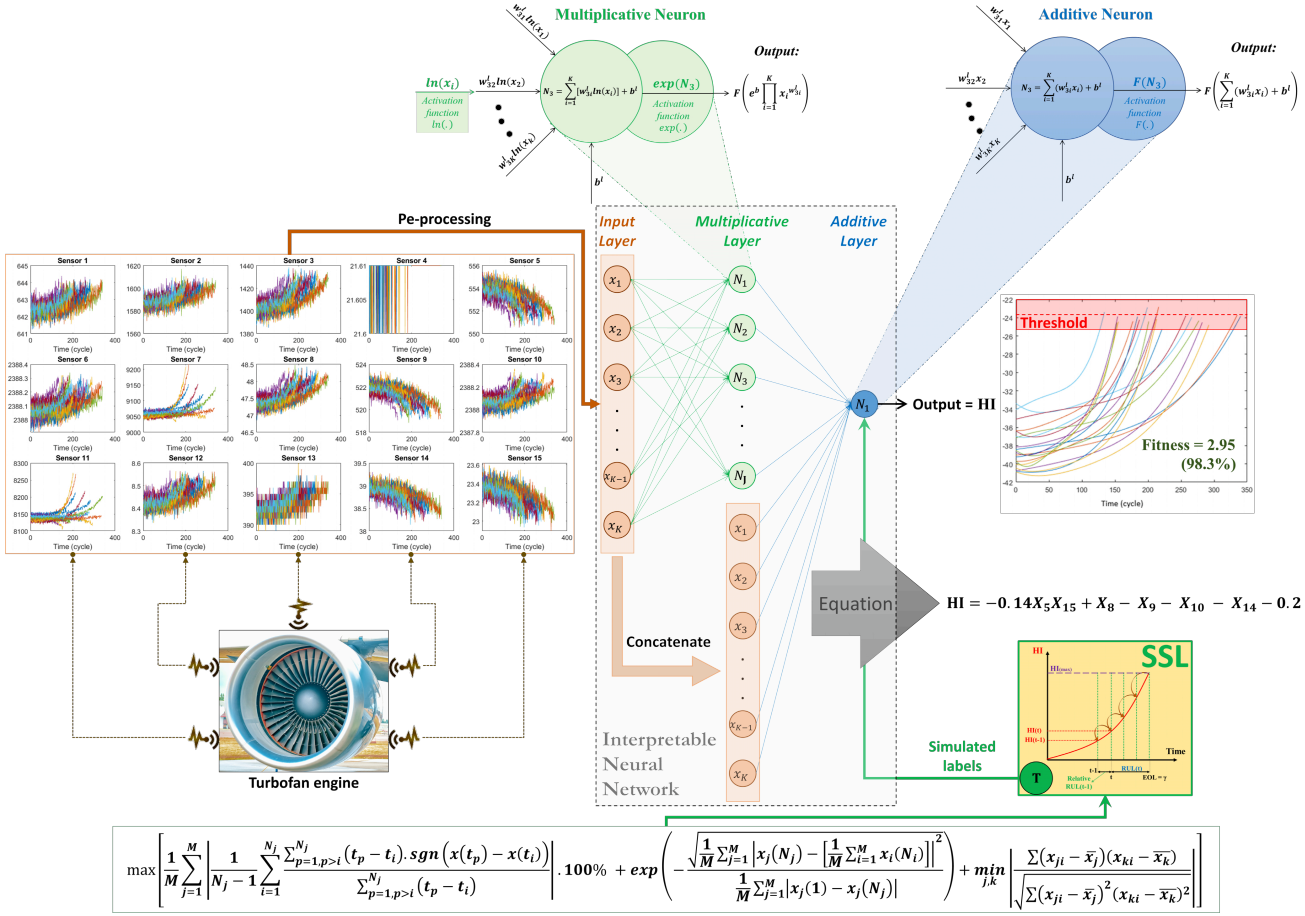
By definition, an ANN is a function approximator that integrates input data into the expected output using a complicated equation. Since an ANN needs large numbers of weights to build acceptable HIs, retrieving the equation is not practical. The number of variables should be dropped while ensuring high levels of performance to achieve an interpretable network that could be transformed into an intuitive and condensed equation expressing a HI. The proposed methodology will demonstrate that combining the weights' discretization—the regulation of their sparsity—and the simultaneous use of both multiplicative and additive neurons results in an interpretability that is satisfactory. The abovementioned ANN can also take into account the physical characteristics that invisibly underlie the components that make up a HI. As a result, the proposed approach may now uncover the ANN's underlying formula, the HI, which properly represents the feature selection and fusion steps.

An additive layer is composed of several typical additive neurons, whereas a multiplicative layer is made up of several multiplicative neurons. Figure A.1 illustrates the proposed framework. First, a multiplicative layer receives the inputs which are either raw sensory data or de-noised ones. Each neuron at this layer is a multiplication of the inputs with various weights and a bias in accordance with Eq. 3.12. There are several multiplication combinations between the inputs when there are plenty of neurons. The output of the multiplicative layer is then added to a subsequent additive layer with a single additive neuron to create the final output. The ANN becomes increasingly complex when more neurons are added to the additive layer, and it is quite probable to overuse a portion of the inputs. Terms that correspond to a single input instead of a combination of them are typically evident once a HI's formula is obtained. For example, if the inputs are x_1, x_2, and x_3, we might get the equation $x_1x_2x_3 + x_1$. It is difficult to establish such an equation utilizing purely the multiplicative layer's outputs to be imported into the subsequent additive layer. Hence, utilizing the inputs of the network in both additive and multiplicative layers is the most applicable architecture. Accordingly, the inputs and outputs from the multiplicative layer are concatenated before being passed into the additive layer. It

is clear that using additive layers and then multiplicative layers results in more complex equations, and the option to multiply the inputs together is missing. For instance, considering four inputs (two by two separated (zero weights) for simplification), the simplified combination of additive+multiplicative yields $(x_1 + x_2)(x_3 + x_4) = x_1x_3 + x_1x_4 + x_2x_3 + x_2x_4$ while the simplified combination of multiplicative+additive yields $x_1x_2 + x_3x_4$, which is less complex. This is due to the ability of the multiplication operation to capture more complex dynamics than the addition one. Consequently, the model should first create all the necessary complex combinations between the features and then decide which of them are important to be added together. It should also be noted that, as mentioned, the option to directly multiply the inputs in the additive+multiplicative configuration, i.e., x_1 and x_2 as well as x_3 and x_4, is missing.

A stream of raw sensor data, a de-noised format, or some extracted features could all be used as the model's inputs. The trained INN serves as the equation for generating the outputs, which are a series of points that make the HI. Before importing the data into the model, a preprocessing step including resampling is required due to the different sequence lengths for each unit. The time-series data points will all be the same length thanks to this step. To do this, there are two techniques. The easier method involves upsampling with interpolation, which involves increasing the number of time-series samples until each sequence is equal to the largest one. Based on the interpolation method selected, those data points are estimated. The second technique involves extending each sequence by adding pseudo-data points since it meets the maximum length needed. This can be performed by padding with an irrelevant value. The padded inputs can then be used to train the model, which outputs the results. This method's sensitive aspect is when it calculates losses. To prevent these pseudo-values from biasing errors through backpropagation, the padded lengths should be perfectly deleted. To continue the next forward pass following the updating of the parameters, the lengths should be padded again. This method eliminates the need for any estimation steps, unlike the first technique. Nevertheless, training time rises remarkably. It was found that both techniques in the present work produce outcomes that are equal, and the first one has been adopted thanks to its simplicity and speed.

So far, the equation for formulating the HI was not completely expressive because the weights could have any real value. Thanks to Eqs. 3.13 to 3.16 for training the INN, the majority of the weights, if not all, shift in the direction of the integers -1, 0, or 1. Weights can converge to the intended values in practice, but they may not always coincide. With this in mind, it is safe to appropriately round the values during validation without compromising accuracy in these circumstances. Utilizing a de-noised version of the sensor data enables all the weights to become ternary (see subsection A.1.3); however, using their raw version does not cause this to happen. As long as the majority of the weights are within the ternary form, a few weights in this last scenario could range from `[-1, 1]`, which could be smoothly rounded to the first decimal point with a trivial accuracy loss. Following training, several non-ternary weights are a result of the noisy raw data. Hence, there is indeed a trade-off between ternarizing the weights and minimizing the E_C loss, which can be controlled by the regularization hyperparameter λ. A large value for λ indicates that more ternary weights have been preferred (better E_R minimization), leading to a more concise equation rather than optimally predicted results (not ideal E_C minimization). Fortunately, the intention is to construct a HI that delivers high criteria scores (Mo, Pr, and



$$\max\left[\frac{1}{M}\sum_{j=1}^{M}\left|\frac{1}{N_j-1}\sum_{i=1}^{N_j}\frac{\sum_{p=1,p>i}^{N_j}(t_p-t_i).sgn\left(x(t_p)-x(t_i)\right)}{\sum_{p=1,p>i}^{N_j}(t_p-t_i)}\right|.100\% + exp\left(-\frac{\sqrt{\frac{1}{M}\sum_{j=1}^{M}\left|x_j(N_j)-\left[\frac{1}{M}\sum_{i=1}^{M}x_i(N_i)\right]\right|^2}}{\frac{1}{M}\sum_{j=1}^{M}\left|x_j(1)-x_j(N_j)\right|}\right) + \min_{j,k}\left|\frac{\sum(x_{ji}-\bar{x_j})(x_{ki}-\bar{x_k})}{\sqrt{\sum(x_{ji}-\bar{x_j})^2(x_{ki}-\bar{x_k})^2}}\right|\right]$$

Figure A.1: Framework of the proposed methodology for interpretability. The inputs are fed into a multiplicative layer, and each neuron applies a multiplication operator. Then, the outputs are concatenated with the inputs and together are driven into the additive layer, and the output is alongside the equation of the constructed HI.

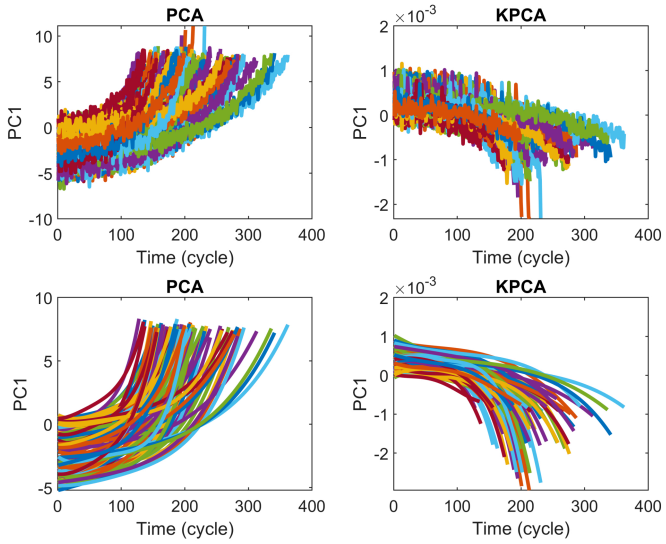


Figure A.2: First principal component of the PCA and KPCA applied to the raw (first row) and de-noised (second row) entire dataset of subset FD001 given 100 engine units.

Tr) rather than relying solely on simulated label values, thus placing greater emphasis on developing concise formulas.

A.1.3 Results and Discussions

In this section, the HIs constructed using the introduced approach are compared and discussed with the outputs of the PCA, KPCA, and GP models for subsets FD001 and FD003, respectively.

Subset FD001

Since the first principal component of the PCA and KPCA covers the largest portion of variations in data, it can be regarded as HI. These PCs were extracted through models applied to the whole dataset including 100 units, which are shown in Figure A.2. Sensor 8 has the highest fitness score for raw inputs, 2.58, which has been improved using the PCA model to 2.85 (10.47%). This value for de-noised inputs increased from 2.91 (sensor 8) to 2.94 (1%). On the other hand, the KPCA model was unable to improve the HI with regard to neither raw nor de-noised inputs, demonstrating that the CMAPSS dataset (especially, subset FD001) has a linear rather than a nonlinear correlation among inputs. As a result, PCA can produce a reasonably appropriate HI for this dataset, and the results argue that complex models for CMAPSS, including DNNs, are unnecessary and redundant. This is also valid for RUL prediction because a better HI results in a more precise RUL forecast. This viewpoint could be supported by the fact that the dataset is an output of a simulation model instead of realistic cases, and that the simulation tool most probably used a number of existing equations in addition to typical noise [2].

The inability to interpret the generated principal components is one of the HI-related drawbacks of the PCA and KPCA methods, as was previously mentioned. Thus, effective solutions to cope with this issue should be introduced and substituted. The following paragraphs provide an overview of the proposed methodology's outcomes.

The developed method constructed the following formula after training with 80% of the de-noised dataset (80 turbofan engines):

$$HI = -0.14X_5X_{15} + X_8 - X_9 - X_{10} - X_{14} - 0.2 \tag{A.1}$$

where X_i denotes the de-noised data from sensor i. Zero weights have been given to the sensors that had no component in the equation, whereas {-1, +1} have been applied to the others. The existence of only one multiplication occurring between the de-noised data shows that only one multiplicative neuron with a bias of $e^b = 0.14$ contributes to the additive layer. The bias of the additive neuron is $b = -0.2$. The INN hyperparameters are listed in Table A.1. It is worth mentioning that all of the hyperparameters were selected via a grid search technique applied to the subset FD001 only, and then the same values were also applied to the subset FD003. The potential values for each hyperparameter are shown in Table A.2. Since a combination of activation functions converts the additive neuron into a multiplicative one, the computational complexity is similar to a vanilla feed-forward NN, i.e. O(L×N2), where L is the number of layers and N is the number of neurons on each layer.

Table A.1: The INN model's hyperparameters.

Dataset	Alpha (α)	Lambda (λ)	Batches	Epochs	Multiplicative Neurons	Additive Neurons	Learning Rate
de-noised	1.6	10^{-5}	4	200	32	1	0.01
raw	1.8	10^{-3}	4	200	64	1	0.01

Table A.2: The hyperparameters' spaces for grid search.

Alpha (α)	[1.1, 1.3, ..., 1.9]	**Multiplicative Neurons**	[1, 16, 32, 64, 128]
Lambda (λ)	[10^{-5}, 10^{-4}, ..., 10^{-2}]	**Additive Neurons**	[1, 16, 32, 64, 128]
Batches	[4, 8, 16]	**Learning Rate**	[10^{-4}, 10^{-3}, 10^{-2}, 10^{-1}]
Epochs	[100, 200, 300]		

The created HIs for each sample of the test data are displayed in Figure A.3(bottom left), demonstrating successful performance for all three metrics (Mo, Pr, and Tr). The overall fitness score is 2.9461, as depicted in Table A.3, demonstrating that the INN effectively combined the de-noised data to produce a greater score for criteria. Whereas the multiplicative layer has a large number of multiplicative neurons (32) in order to obtain a concise formula, as expected, only one multiplicative neuron—the one that multiplies the features X_5 and X_{15}—contributes to the output after implementing weight regularization with sparsity control.

Given directly unprocessed raw data as input, the suggested model produces the following equation:

$$HI = 0.04\frac{X_1^{0.4}X_2^{0.3}X_6^{0.2}X_7^{0.1}X_{12}^{0.1}}{X_5^{0.2}X_{14}^{0.3}X_{15}^{0.2}} - X_5 + X_8 - X_9 + X_{11} + 0.11 \tag{A.2}$$

where X_i denotes the raw data from sensor i. Since it is more challenging to achieve an effective equation given raw noisy data than de-noised ones if only the ternary format of the weights is employed, the HI equation contains more terms, as expected. In order to construct Eq. A.2, some weights of the multiplication layer had to be float values that

A

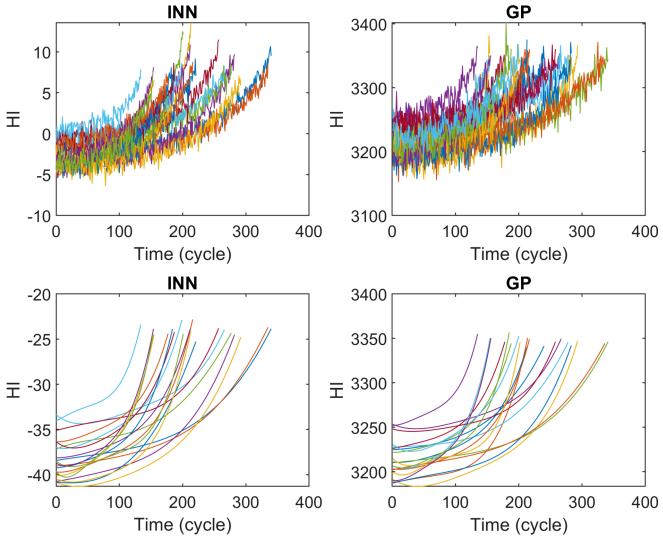


Figure A.3: HIs constructed by the proposed (INN) and two-stage GP (GP) models, utilizing raw (first row) and de-noised (second row) data for 20 test engine units of subset FD001.

were rounded to the closest first decimal point. Figure A.3(top left) displays the created HIs for each test unit. As can be seen in Table A.3, the fitness score for the raw data is 2.7407, which is lower than the de-noised form. Once more, the proposed INN was able to effectively combine the raw data to generate a better HI in terms of criteria and interpretability. The number of neurons in the multiplicative layer was doubled (Table A.1)

Table A.3: Scores for HI evaluation criteria of PCA, KPCA, GP, and the proposed model (INN), all trained on 80 engine units from subset FD001, calculated considering the 20 test engine units.

		PCA	KPCA	GP	Proposed model	Best Sensor (S 8)
Monotonicity	Raw	0.99	0.96	0.97	0.98	0.96
	Denoised	1.00	1.00	1.00	1.00	1.00
Trendability	Raw	0.92	0.64	0.83	0.92	0.78
	Denoised	0.96	0.95	0.97	0.99	0.97
Prognosability	Raw	0.96	0.71	0.92	0.83	0.87
	Denoised	0.97	0.70	0.97	0.96	0.95
Fitness	Raw	2.87	2.31	2.73	2.74	2.61
	Denoised	2.94	2.65	2.93	2.95	2.91
* "Green color ⟶ Red color" equalizes "Best result ⟶ Worst result"					*midpoint*	

Table A.4: Scores for HI evaluation criteria of PCA, KPCA, GP, and the proposed model (INN), all trained on 80 engine units from subset FD001, calculated considering both the 80 training and 20 test units.

		PCA	KPCA	GP	Proposed model	Best Sensor (S 8)
Monotonicity	Raw	0.99	0.97	0.98	0.99	0.97
	Denoised	1.00	1.00	1.00	1.00	1.00
Trendability	Raw	0.93	0.60	0.79	0.90	0.74
	Denoised	0.97	0.95	0.97	0.99	0.97
Prognosability	Raw	0.93	0.65	0.90	0.81	0.87
	Denoised	0.97	0.68	0.96	0.96	0.94
Fitness	Raw	2.85	2.22	2.67	2.70	2.58
	Denoised	2.94	2.63	2.93	2.94	2.91
* "Green color ⟶ Red color" equalizes "Best result ⟶ Worst result"					*midpoint*	

since processing raw data is more complicated. This makes training more complex, but the sparsity control once again eliminates the unneeded weights to still generate a simple formula. The hyperparameters α and λ have to be raised to increase the zeroed weights and emphasize this process more, respectively, so that this doubling of the neurons can be compensated. For comparison, the outputs of the most recent study (a two-stage GP model [3]) are shown in Figure A.3(right), along with the outputs of the proposed approach. It should be emphasized that although the equation produced from the two-stage GP model was only applied to and resulted from de-noised data, the same constructed equation was also used to generate HIs given the raw data in order to make a comparison. Tables A.3 and A.4 present, respectively, the evaluation criteria scores for the test set and for the whole dataset (100 units). The latter is because the prognostic model, which also needs to be trained on only the training portion, could achieve more accurate RUL predictions for the test portion when the criteria scores for the entire HIs, including the training and test portions, are high, rather than only for the test HIs.

The developed model using the de-noised data has the best fitness score of all (2.95). PCA-based HI has a close fitness score (2.94), but the derived HI equation is complicated to comprehend. The GP model similarly achieves a high score (2.93); however, the authors [3]) only took into account the highest-quality inputs (based on the feature extractor in the first stage) in the second stage (which has been dedicated to the feature fusion process). It should be emphasized that a larger INN including more layers and neurons with decreasing α could have produced even higher fitness scores, but with less interpretability and more complex functions. The results show that the developed methodology is superior for the subset FD001 as a result of the highest fitness score and, in the meantime, good interpretability.

Subset FD003

The PCA- and KPCA-based HIs for subset FD003, considering the whole dataset, are shown in Figure A.4. Similar to the subset FD001, sensor 8 has the highest fitness score for raw inputs, 2.56, which has been diminished using the PCA model to 2.29 (-10.55%). This value for de-noised inputs decreased from 2.80 (sensor 8) to 2.47 (-11.79%). In contrast to the subset FD001, the KPCA model provides slightly better HIs compared to the PCA model, with scores of 2.37 and 2.49 for raw and de-noised inputs, respectively. However, they are still less than the best input (sensor 8).

The INN model constructed the following equation after training with 80% of the de-noised dataset:

$$HI = -3.04\frac{X_5X_8X_9X_{10}X_{16}}{X_{14}X_{15}} - 1.51 \tag{A.3}$$

where X_i denotes the de-noised data from sensor i. In comparison to Eq. A.1 for the subset FD001, although the equation format has changed, the same sensors are involved, except for sensor 16 (X_{16}) which was the different sensor used in subset FD003. Although the same sensors were included because the objects (engines) are of a special model, the new sensor 16 was also included in the formula, possibly because the subset FD003 contains engines that have two failure modes (rather than one), and thus this sensor likely carries the information related to the failure modes. As can be seen, mainly multiplication neurons contribute to a bias of the additive neuron, which is $b = -1.51$. It should be mentioned

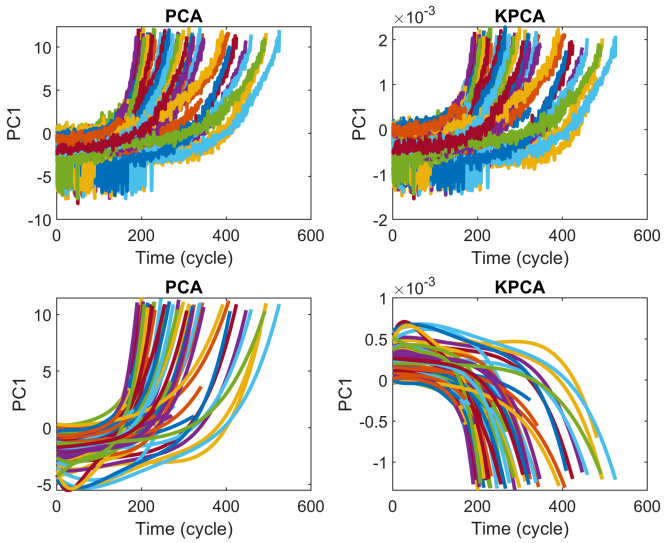


Figure A.4: First principal component of the PCA and KPCA applied to the raw (first row) and de-noised (second row) entire dataset of subset FD003 given 100 engine units

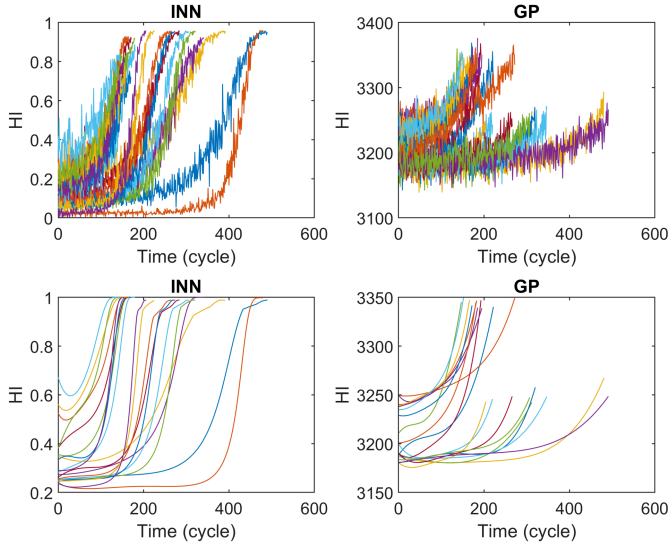


Figure A.5: HIs constructed by the proposed (INN) and two-stage GP (GP) models, utilizing raw (first row) and de-noised (second row) data for 20 test engine units of subset FD003.

that the INN hyperparameters for the subset FD003 are the same as the FD001 (Table A.1) since the purpose is also the interpretability rather than purely the high criteria scores. The created HIs for each sample of the test data are displayed in Figure A.5(bottom left), demonstrating successful performance for all three metrics. The overall fitness score is 2.8624, as depicted in Table A.5, confirming the INN's performance.

Table A.5: Scores for HI evaluation criteria of PCA, KPCA, GP, and the proposed model (INN), all trained on 80 engine units from subset FD003, calculated considering the 20 test engine units.

		PCA	KPCA	GP	Proposed model	Best Sensor (S 8)
Monotonicity	Raw	0.99	0.98	0.96	0.99	0.97
	Denoised	1.00	1.00	0.99	1.00	1.00
Trendability	Raw	0.90	0.88	0.58	0.91	0.78
	Denoised	0.98	0.98	0.93	0.87	0.97
Prognosability	Raw	0.59	0.63	0.61	0.90	0.92
	Denoised	0.59	0.63	0.60	0.99	0.94
Fitness	Raw	2.47	2.50	2.15	2.80	2.67
	Denoised	2.57	2.60	2.52	2.86	2.91
* "Green color ⟶ Red color" equalizes "Best result ⟶ Worst result"					*midpoint*	

Table A.6: Scores for HI evaluation criteria of PCA, KPCA, GP, and the proposed model (INN), all trained on 80 engine units from subset FD003, calculated considering both the 80 training and 20 test units.

		PCA	KPCA	GP	Proposed model	Best Sensor (S 8)
Monotonicity	Raw	0.99	0.99	0.96	0.99	0.98
	Denoised	1.00	1.00	0.99	1.00	0.99
Trendability	Raw	0.75	0.75	0.43	0.86	0.68
	Denoised	0.92	0.87	0.73	0.83	0.86
Prognosability	Raw	0.56	0.63	0.60	0.89	0.90
	Denoised	0.56	0.62	0.61	1.00	0.94
Fitness	Raw	2.29	2.37	1.99	2.74	2.56
	Denoised	2.47	2.49	2.33	2.82	2.80
* "Green color ⟶ Red color" equalizes "Best result ⟶ Worst result"					*midpoint*	

Given directly unprocessed raw data of the subset FD003 as input, the suggested model produces the following formula:

$$HI = -0.024 \frac{X_1^{0.2} X_2^{0.2} X_5^{0.1} X_6^{0.3} X_7^{0.2} X_8^{0.2} X_9^{0.3} X_{11}^{0.1} X_{12}^{0.3} X_{14}^{0.3} X_{16}^{0.2}}{X_{14}^{0.1} X_{15}^{0.1}} - 0.1(X_1 + X_2 - X_6) - 3.29 \tag{A.4}$$

where X_i denotes the raw data from sensor i. Similar to the subset FD001, to construct Eq. A.4 for the noisy data, some weights of the multiplication layer had to be float values that were rounded to the closest first decimal point. Figure A.5(top left) displays the created HIs for each test unit. As can be seen in Table A.5, the fitness score for the raw data is 2.7893, which is lower than the de-noised form. Again, the suggested INN was successful in combining the raw data to produce a better HI in terms of criteria and interpretability. Training becomes more complicated as a result, but the sparsity control once more excludes the extra weights to produce a straightforward equation.

In Figure A.5, the outputs of the INN and the two-stage GP model [3] are displayed for comparison. It is important to note that, even though the equation generated by the two-stage GP model was the result of applying to only the de-noised data of the subset FD001, the same built equation was also used to obtain HIs for the subset FD003 in order to compare results. Tables A.5 and A.6 present, respectively, the evaluation metrics scores for the test set and for the whole dataset. Regarding the latter, it should be again emphasized that if the criteria scores for the complete set of HIs (both the training and test) are high, it will increase the accuracy of RUL predictions for the test portion.

The INN model using the de-noised data has the highest fitness score (2.86) compared to the other models. PCA and KPCA models generated HIs with lower fitness scores,

A

and the extracted HI formula is also complicated to understand. The GP model achieves the lowest score; however, the used equation has been extracted from only the subset FD001. The findings demonstrate that the INN model performs better because of its high fitness score and, concurrently, its high interpretability, i.e., the equation of the built HI can be interpreted and is readable in terms of the inputs. In contrast to the thousands of parameters in typical DL models, the number of parameters in HI equations of INN, given de-noised inputs, is 8 and 9, and given raw inputs, it is 14 and 18, for subsets FD001 and FD003, respectively.

Health indicator threshold

Since the HI labels were simulated according to a SSL framework with an initial range of [0–1] [4], which can be scaled in any desired range, like [0–10], the end limit was already considered as the threshold of HI for the simulated labels. However, the true threshold of the designed HI for the Time-To-Failure should be selected (or calculated) after the training phase, considering the constructed HIs rather than the simulated HIs. With this in mind, the mean value of the constructed HIs at the EoL or at a predefined level based on the intended reliability and safety as well as uncertainty could be selected as the threshold. For example, the range of the simulated labels has been scaled to [0–10] for the subset FD001, while the range for subset FD003 is [0–1] without any scaling, which proves that the employed SSL model is insensitive to the range of labels and only the pattern of targets is important. However, as explained, the constructed HIs are the basis for determining the threshold. With this in mind, for subset FD001, the threshold considering the raw data-based constructed HI is 8.56 and the threshold considering the de-noised data-based constructed HI is -24.04 according to the mean value of the predicted HIs in the training phase. Similarly, these thresholds for subset FD001 considering a higher safety confidence, 5% earlier than the EoL, are 6.09 and -26.56 for the raw data-based and de-nosied data-based constructed HIs, respectively. These threshold values for both subsets in comparison with the testing phase in order to calculate the error between them are shown in Table A.7. The error is calculated as (Threshold$_{Training}$ - Threshold$_{Test}$)/Threshold$_{Test}$, to consider the test as the basis. The negative errors indicate safer thresholds, while positive ones indicate earlier failures. It should be noted that the true thresholds (at 100% × EoL) for the test units are always greater (later) than the thresholds at 95% × EoL based on the training units, which in turn shows that with a 5% interval, the determined threshold is almost safe.

Table A.7: Mean value of the constructed HIs at the EoL (100% × EoL) and 5% earlier than the EoL (95% × EoL) to determine the threshold for the Time-To-Failure, which should be based on the training phase.

	Inputs	Threshold at 95% × EoL			Threshold at 100% × EoL		
		Training	Test	Error (%)	**Training**	Test	Error (%)
Subset FD001	Raw data	**6.09**	5.70	6.84	**8.56**	8.55	0.12
	De-noised data	**-26.56**	-26.68	-0.45	**-24.04**	-23.95	0.38
Subset FD003	Raw data	**0.84**	0.85	-1.18	**0.88**	0.90	-2.22
	De-noised data	**0.97**	0.99	-2.02	**0.98**	0.99	-1.01

The INN's ability to directly convolve inputs together is the main reason why it outperforms other compared methods, which lack the ability to multiply inputs, in terms of both higher evaluation scores for HIs and interpretability. Furthermore, sparsity control

through the use of discretized (ternary) weights has improved interpretability by making the neural network more compact, resulting in more concise output equations.

A.1.4 Conclusions

Designing an appropriate HI that satisfies the evaluation requirements of Mo, Pr, and Tr for prognostics while also being interpretable for an engineering system or structure in PHM is a challenging task. The proposed methodology in this section, INN, has the potential to combine the sensory data and create the intended HI. This section showed the potential of the INNs to achieve ultimate performances by making their prohibitively large equations compact and readable. As such, the HI function has also been simplified by combining the multiplicative and additive neurons with the discretized weights employing sparsity control. It was shown that even when increasing the number of neurons, the extracted equation is still constructed only by the contributions of a bunch of neurons by controlling the hyperparameters α and λ. The results demonstrated that the proposed methodology in this section is superior based on its combined highest score and interpretability.

A

A.2 HIs using Acoustic Emission

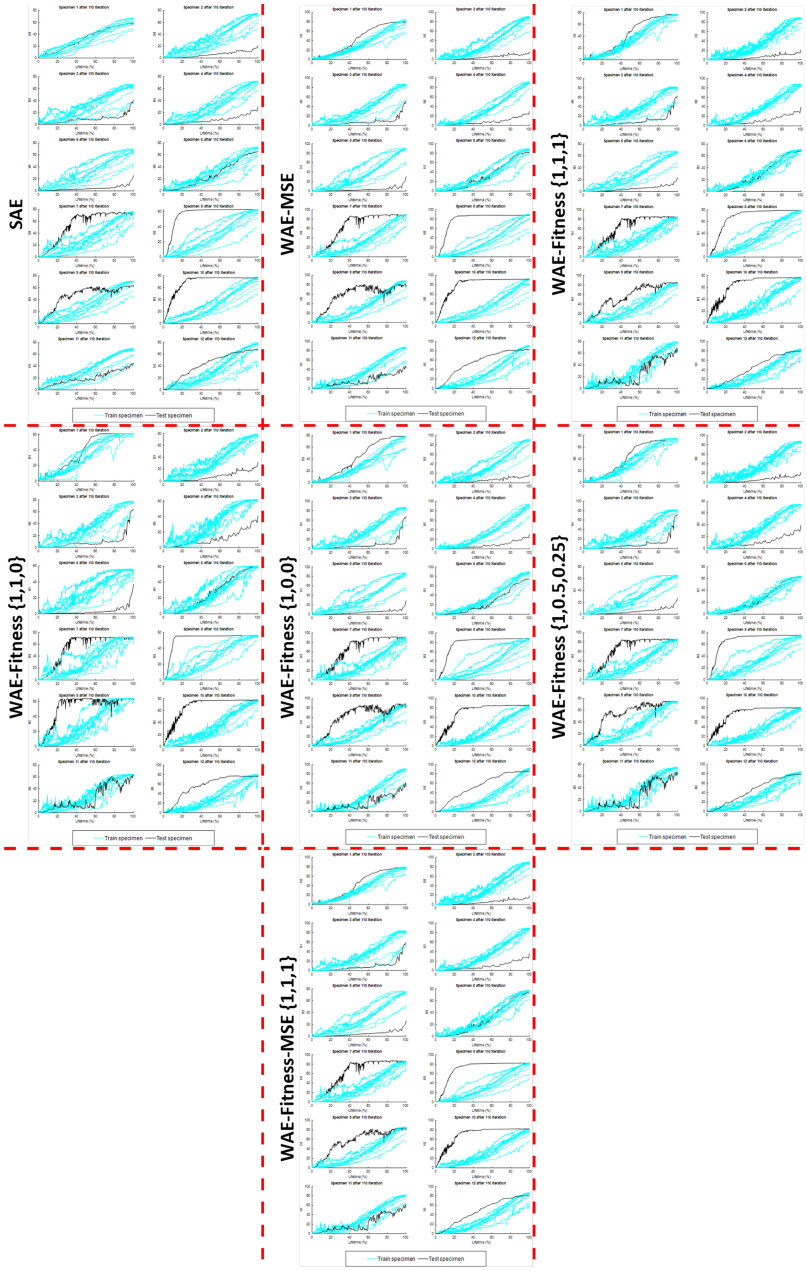


Figure A.6: HIs constructed by different ensemble learning methods applied to the 2^{nd} framework (with A.1 PCA version implemented) utilizing acoustic emission data.

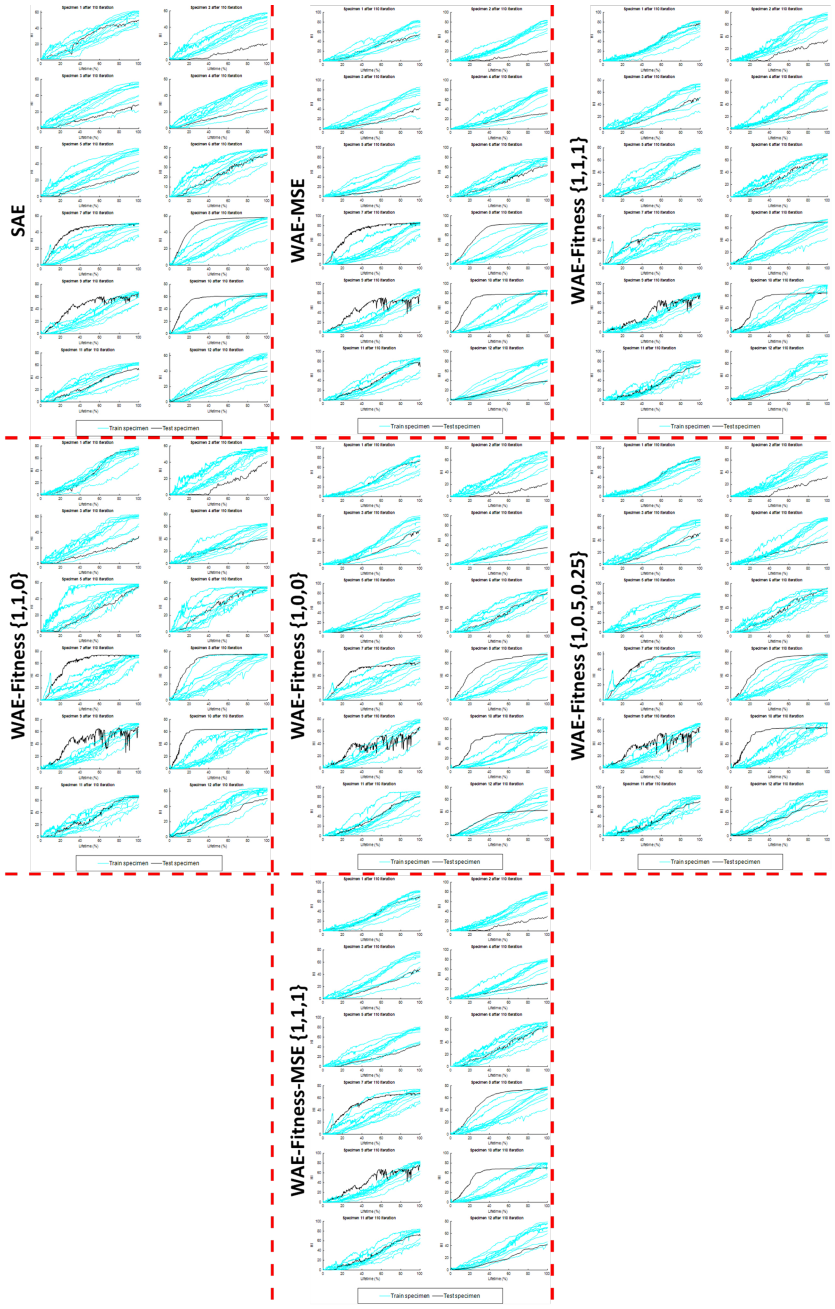


Figure A.7: HIs constructed by different ensemble learning methods applied to the 2^{nd} framework (with B.1 PCA version implemented) utilizing acoustic emission data.

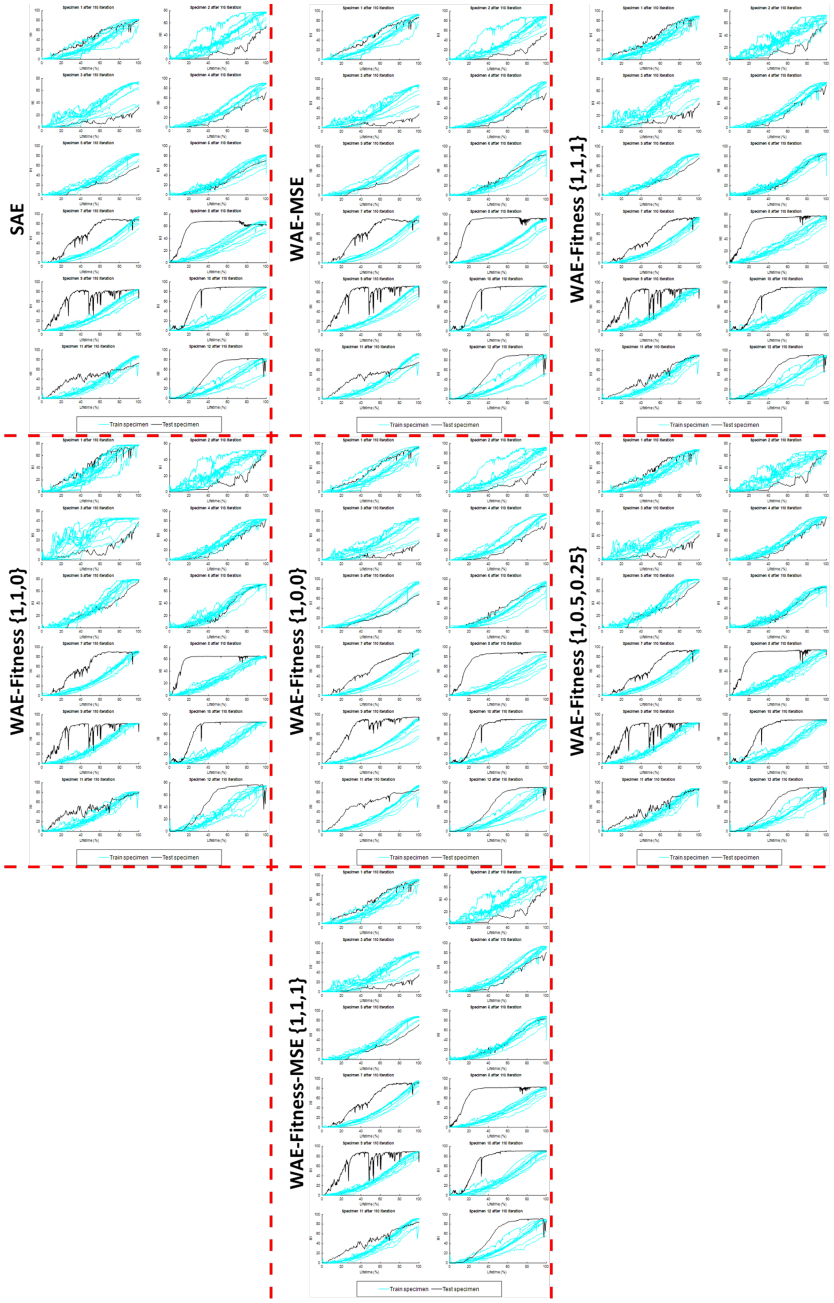


Figure A.8: HIs constructed by different ensemble learning methods applied to the 2^{nd} framework (with A.2 PCA version implemented) utilizing acoustic emission data.

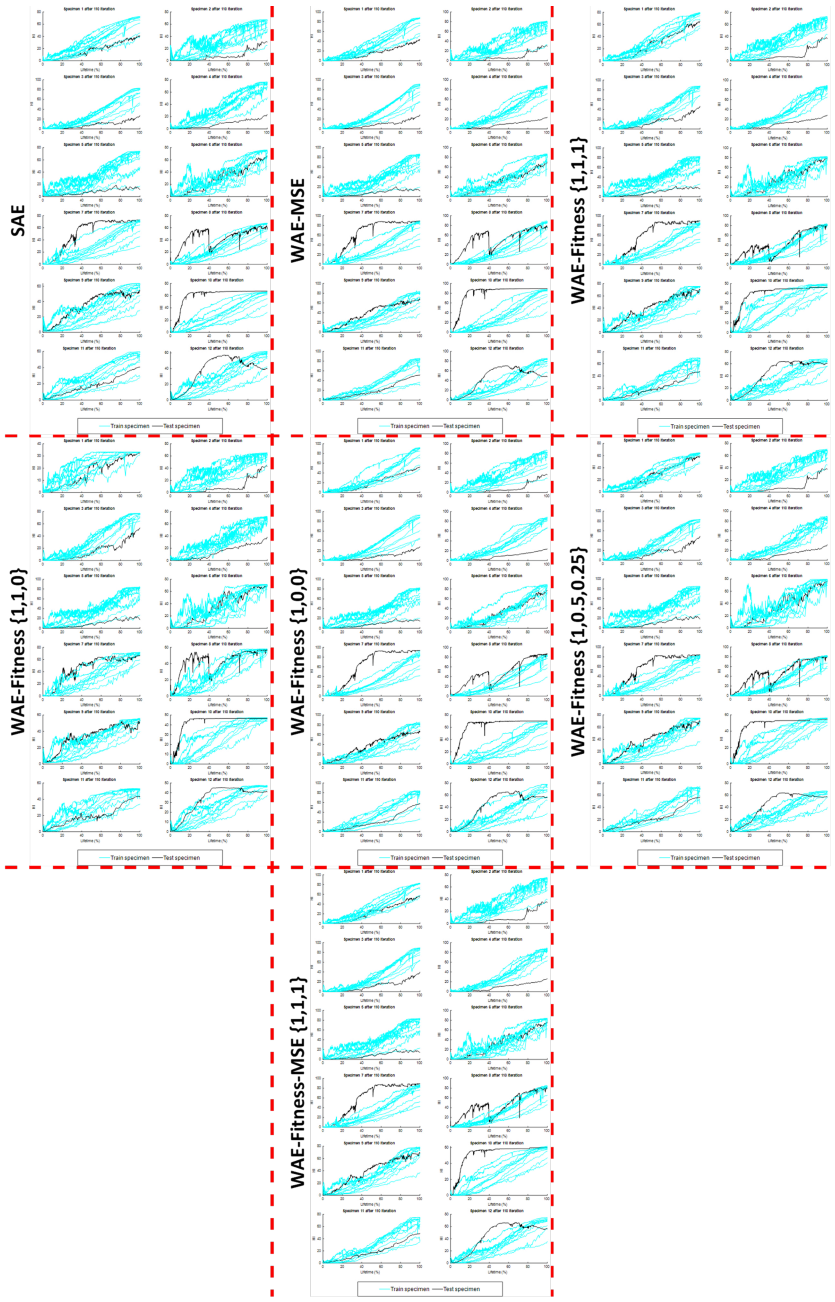


Figure A.9: HIs constructed by different ensemble learning methods applied to the 2^{nd} framework (with B.2 PCA version implemented) utilizing acoustic emission data.

A

A.3 HIs using Guided Waves

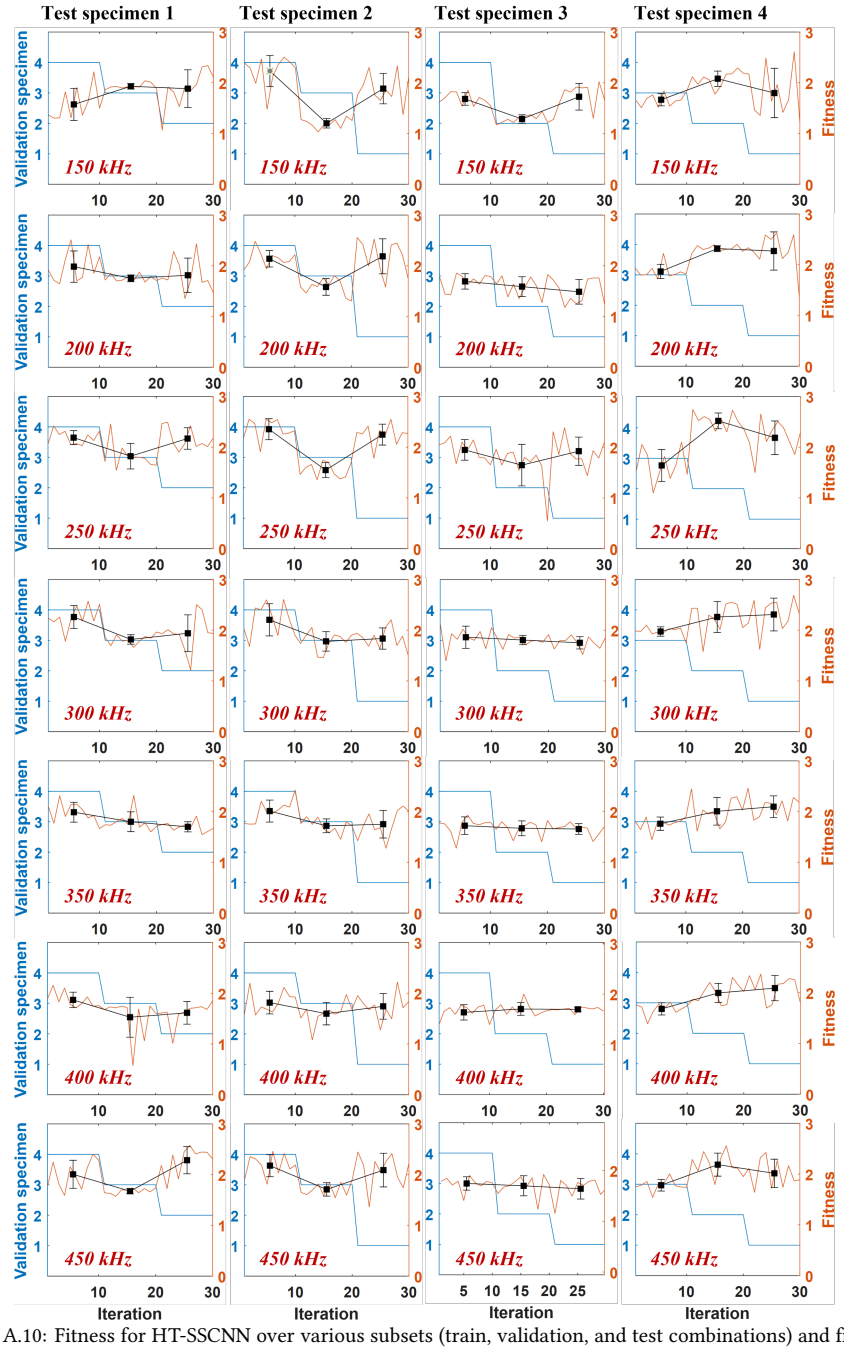


Figure A.10: Fitness for HT-SSCNN over various subsets (train, validation, and test combinations) and frequencies of GW data in the NASA dataset-layup 1 (fitness calculated from Eq. 3.4, considering all units referred to as 'F-All').

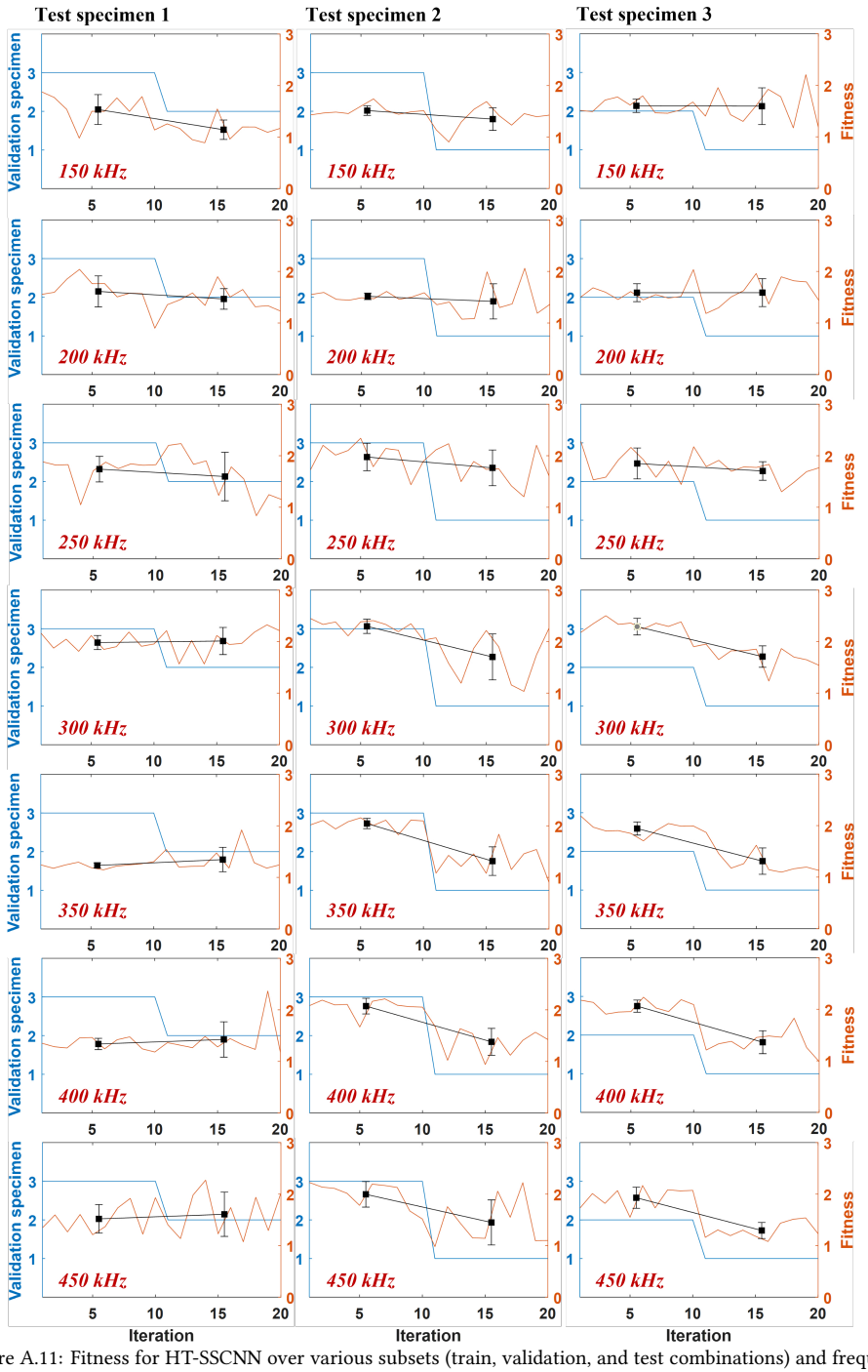


Figure A.11: Fitness for HT-SSCNN over various subsets (train, validation, and test combinations) and frequencies of GW data in the NASA dataset-layup 2 (fitness calculated from Eq. 3.4, considering all units referred to as 'F-All').

A

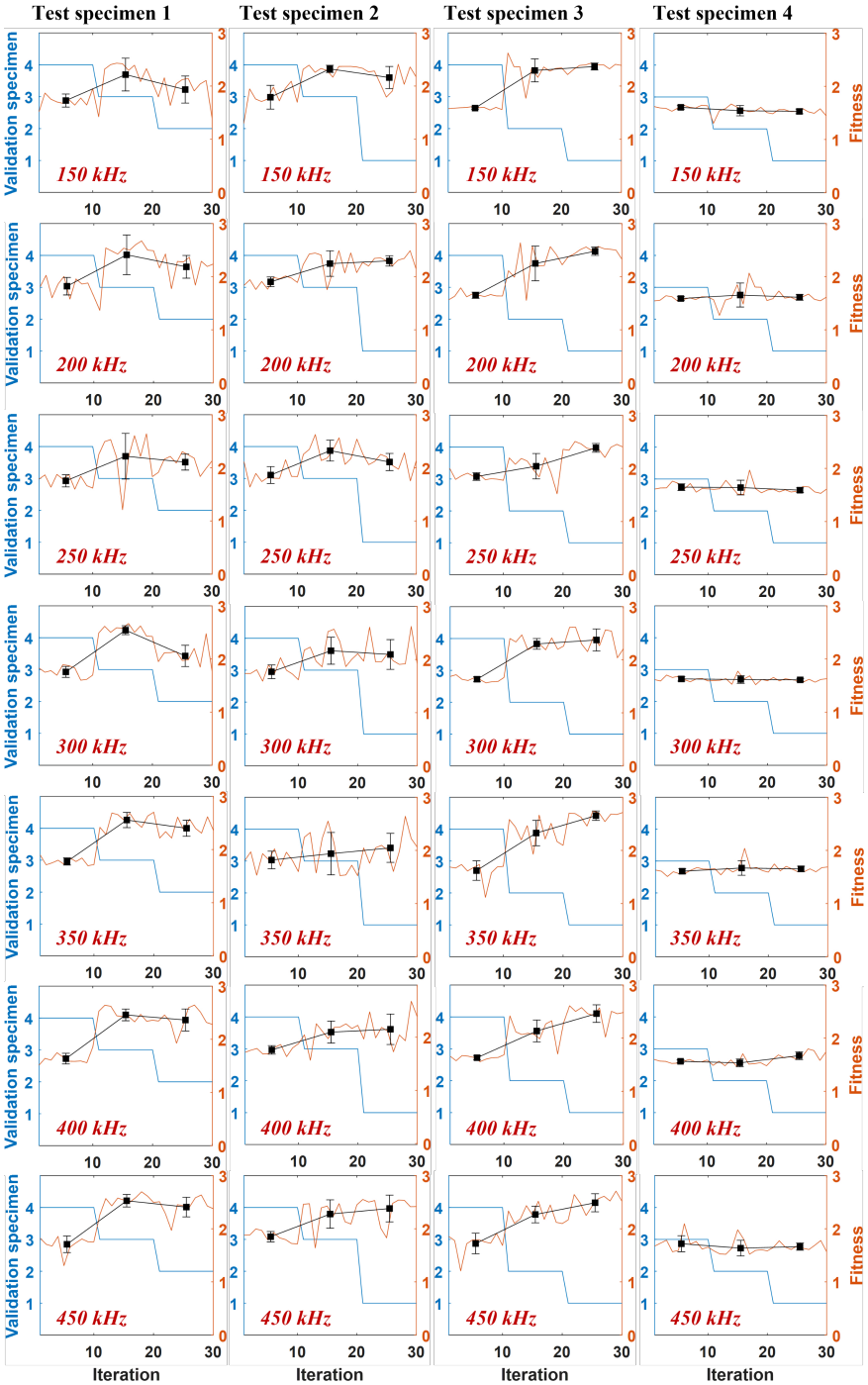


Figure A.12: Fitness for HT-SSCNN over various subsets (train, validation, and test combinations) and frequencies of GW data in the NASA dataset-layup 3 (fitness calculated from Eq. 3.4, considering all units referred to as 'F-All').

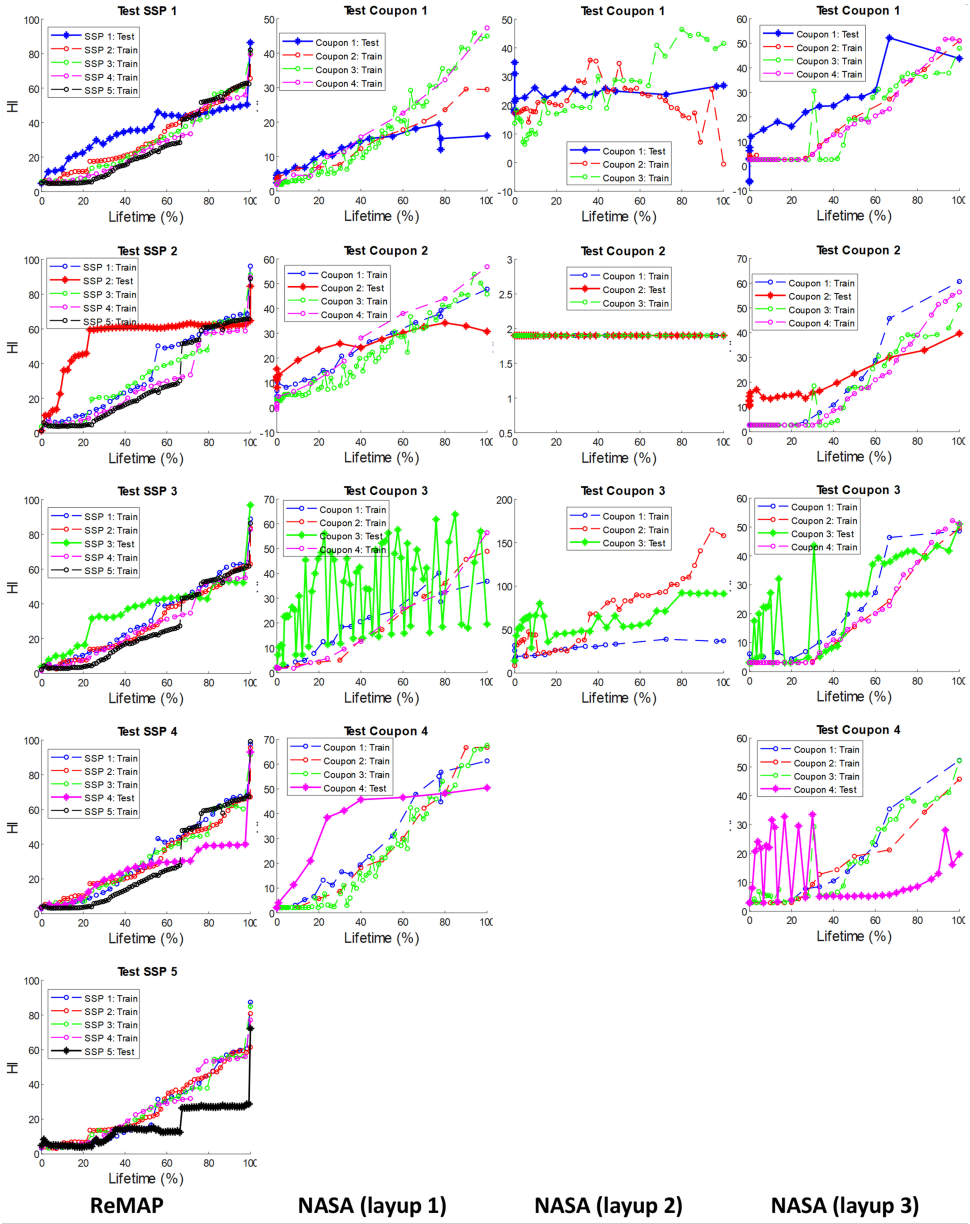


Figure A.13: HIs obtained by the proposed framework upon GW data with FC-EL for different datasets, given all frequency inputs.

A

References

[1] E. Ramasso and A. Saxena, "Review and analysis of algorithmic approaches developed for prognostics on cmapss dataset," in *Annual Conference of the Prognostics and Health Management Society 2014.*, 2014.

[2] M. Moradi, P. Komninos, R. Benedictus, and D. Zarouchas, "Interpretable neural network with limited weights for constructing simple and explainable hi using shm data," in *Annual Conference of the PHM Society*, vol. 14, 2022.

[3] K. T. Nguyen and K. Medjaher, "An automated health indicator construction methodology for prognostics based on multi-criteria optimization," *ISA transactions*, vol. 113, pp. 81–96, 2021.

[4] M. Moradi, A. Broer, J. Chiachío, R. Benedictus, T. H. Loutas, and D. Zarouchas, "Intelligent health indicator construction for prognostics of composite structures utilizing a semi-supervised deep neural network and shm data," *Engineering Applications of Artificial Intelligence*, vol. 117, p. 105502, 2023.

Acknowledgments

The completion of this thesis marks the end of an extraordinary journey that spanned three and a half years and was filled with cherished memories that seem to have passed in the blink of an eye. I extend my heartfelt gratitude to all those who have contributed to this unforgettable experience.

First and foremost, I am grateful to my supervisor, Dr. Dimitrios Zarouchas. Dimitrios, your initial trust in accepting me as a PhD candidate, as well as your insightful guidance during crucial times, were incredibly valuable. As always, I say that your feedback has been painful yet impactful, aligning perfectly with my expectations. Our debates have helped me become more mature, and I consider myself extremely fortunate to have learned from your wisdom and scholarly expertise. Furthermore, your sense of humor has made our interactions more enjoyable. Σ' ευχαριστώ!

I also extend my gratitude to Prof. Rinze Benedictus, my promotor, whose constructive feedback has not only shaped my PhD journey but also influenced my personal vision and future career. Rinze, your visionary approach to academia is commendable, and I am grateful for your trust in me. Dank u wel!

To my co-supervisor, Dr. Juan Chiachío, I am thankful for your support and insightful feedback. Your guidance has been highly valuable to my progress. Gracias!

Gemma, your assistance in handling endless bureaucratic tasks has been invaluable. You have always been there whenever I have needed help.

Special recognition is owed to Dr. Manuel Chiachío at the University of Granada, the dreamy coordinator of the ENHAnCE project. Manuel, your inspiring proposal seven years ago for the ITN project laid the foundation for this remarkable journey. I am deeply grateful for everything. Additionally, I extend my heartfelt thanks to Maria for her tireless efforts in managing various aspects of the project, ensuring the success of our training weeks. Lastly, I express my gratitude to Rafael, our mentor in management and leadership, for his steadfast support.

To my dear colleagues and friends from the ENHAnCE project—Shankar, Wen, Juan, Tianzhi, Javier, Aravind, Tasdeeq, Amond, Ali, and Mohammad Ali—your invaluable friendship and collaborative efforts have enriched our project. Each of you has made significant contributions. Shankar, our profound discussions spanning science, history, and politics have offered enlightening perspectives. I eagerly anticipate witnessing your bouldering sessions with your daughter next time. Wen, your infectious smile has consistently brought light to even our tiring times. Thank you all for your dedication to this project.

Many ups and downs marked my journey since commencing my PhD in the Netherlands, from dealing with COVID to coping with tough times back home while I was thousands of miles away. In those difficult moments, my friends were my rock, offering support. My friend Ramin, for his support and friendship. I am fortunate to have crossed paths with you in Delft, and I treasure our friendship. I do not forget all the countless memories

we have shared since the first online meeting that we set up even before arriving in the Netherlands. Ahmed and Parvin, thank you for always being there for me. Ahmed, our random encounter on my second flight to Amsterdam marked the start of a meaningful friendship. Parvin, your help with a particularly special matter means a lot to me. Together, we have shared laughter and created memories that will last a lifetime. To the rest of my friends in the Netherlands—Amir, Hamid, Farhad, Mozhgan, and Marieke—your camaraderie has enriched my experience abroad. Each moment spent in your company has been a cherished memory.

Milad, a true role model, exemplifies hard work and dedication. You tirelessly pursue your research without complaint or excuse. From the very beginning, even during the challenging times of COVID, you have been a constant source of support, both personally and professionally. Rose, a diligent researcher, is someone I deeply admire. I have no doubt you will emerge as a prominent scientific figure in the future. Your kindness and vibrant energy have illuminated even the darkest days. Whether it is our adventures outside of work or our heartfelt conversations during office hours, your friendship has brought me immense joy.

To my office mates—Wenjie, Stratos, Thanos, Lubin, Nicolas, Chantal, Jimmy, Agneta—your presence has turned our workplace into a hub of friendship. Wenjie, your politeness and respect set the tone for all of us. Stratos, your humor always lightens the mood. Thanos, your warmth and support have been a real comfort. It's always a pleasure spending time with you. Lubin, your hard work and respect for others are qualities I admire.

To my fellow PhDs and postdocs—Xi, Agnes, Sasan, Ali, Majid, Amin, Panos, Nan, Silvia, Ferda, Thanasis, Davide, Daniel, George, Deniz, Mirko, Ran, Mariana, Onur, Eva—thank you for your friendship and support, which made this journey unforgettable. Your assistance and shared fun moments enriched my experience.

I extend my gratitude to other colleagues and professors—Vahid, Calvin, Sofia, Nick, Nan, John, Baris, Rene, and Julie—for their mentorship and guidance. Each of you has left an indelible mark on my academic journey. Vahid, for his warmth in conversation and expertise in AI; Calvin, whose engaging lectures sparked my passion for teaching.

To my friends in my homeland, Iran, and other countries—Amin, Mohsen, Behnam, Reza, Ali, Masoud, and others—your unwavering support has been a source of strength throughout my journey.

Finally, I extend my deepest gratitude to my family, the bedrock of my journey. To my dear friend turned fiancée, Parastou, thank you for your boundless love and unwavering support. You have changed my perspective on life. Our shared journey, beginning with our arrival in the Netherlands amidst the challenges of COVID, has been a testament to our resilience. To my brother, Mojtaba, thank you for always being there for me. To my parents, Maliheh and Yadollah, your unconditional love, sacrifices, and endless encouragement have propelled me forward in pursuit of my dreams. Thank you all.

هرچه آموختم در مکتب عشق شما آموختم و هرچه بکوشم قطره ای از دریای بی کران مهربانیتان را سپاس نتوانم بگویم. دوستتان دارم.

Morteza
Delft, January 2024

Curriculum Vitæ

Morteza Moradi was born in Isfahan, the city renowned for its Persian architecture and history in central Iran, in 1992. He received his B.Sc. degree in mechanical engineering from the University of Kashan, Isfahan, Iran, in 2014, and the M.Sc. degree (with honors) in aerospace engineering from the Iran University of Science and Technology (IUST), Tehran, Iran, in 2018. From 2018 to 2020, he served as a research assistant at the Nondestructive Testing and Condition Monitoring Laboratory at IUST. During the same period, from 2012 to 2020, he worked part-time and full-time in various roles within the mechanical and manufacturing sectors of different companies, including as a design engineer, quality control officer, project control officer, and project control manager.

In 2020, he joined the Structural Integrity and Composites (SI&C) section and the Center of Excellence in Artificial Intelligence for Structures, Prognostics & Health Management, at Delft University of Technology (TU Delft) in Delft, the Netherlands, where he has been pursuing his Ph.D. degree under the supervision of Dr. D. Zarouchas as daily supervisor, Dr. J. Chiachío as co-supervisor, and Prof. Dr. R. Benedictus as promoter. He was granted funding through the Marie Sklodowska-Curie grant by the European Union's Horizon 2020 research and innovation program.

Morteza has been actively serving as a technical reviewer for a number of scientific journals that are released by reputable publishers such as Elsevier, IEEE, Taylor & Francis, and SAGE Publications. His areas of interest in research are artificial intelligence (AI), information fusion, signal processing, structural health monitoring (SHM), non-destructive testing (NDT), prognostics and health management (PHM), thermography, acoustic emissions, composite structures, and finite element modeling (FEM).

List of Publications

Journal papers

[1] **M. Moradi** and D. Zarouchas. *Multi-sensor fusion for aerospace structural health indicators: intelligent integration of passive and active SHM sensory networks.* Information Fusion, (In preparation), 2024.

[2] S. Galiana, **M. Moradi**, P. Wierach, and D. Zarouchas. *Innovative welding integration of acousto-ultrasonic composite transducers onto thermoplastic composite structures.* Structural Health Monitoring, 14759217241247766, 2024.

[3] **M. Moradi**, F.C. Gul, and D. Zarouchas. *A novel machine learning model to design historical-independent health indicators for composite structures.* Composites Part B: Engineering, 275:111328, 2024.

[4] **M. Moradi**, P. Komninos, and D. Zarouchas. *Constructing explainable health indicators for aircraft engines by developing an interpretable neural network with discretized weights.* Applied Intelligence, (Under review), 2023.

[5] **M. Moradi**, J. Chiachío, and D. Zarouchas. *Intelligent health indicator extraction for single T-stiffener composite panels: a two-stage machine learning approach with adaptive standardization and physics-based Bayesian optimization.* Engineering Applications of Artificial Intelligence, (Under review), 2023.

[6] **M. Moradi**, G. Galanopoulos, T. Kuiters, and D. Zarouchas. *A novel intelligent health indicator using acoustic waves: CEEMDAN-driven semi-supervised ensemble deep learning.* Mechanical Systems and Signal Processing, (Under review), 2023.

[7] **M. Moradi**, A. Broer, J. Chiachío, R. Benedictus, T. H. Loutas, and D. Zarouchas. *Intelligent health indicator construction for prognostics of composite structures utilizing a semi-supervised deep neural network and shm data.* Engineering Applications of Artificial Intelligence, 117:105502, 2023.

[8] **M. Moradi**, R. Ghorbani, S. Sfarra, D. M. Tax, and D. Zarouchas. *A spatiotemporal deep neural network useful for defect identification and reconstruction of artworks using infrared thermography.* Sensors, 22(23):9361, 2022.

[9] **M. Moradi** and S. Sfarra. *Rectifying the emissivity variations problem caused by pigments in artworks inspected by infrared thermography: A simple, useful, effective, and optimized approach for the cultural heritage field.* Infrared Physics & Technology, 115:103718, 2021.

[10] B. Ameri, **M. Moradi**, and R. Talebitooti. *Effect of honeycomb core on free vibration analysis of fiber metal laminate (FML) beams compared to conventional composites.* Composite Structures, 261:113281, 2021.

[11] B. Ameri, **M. Moradi**, B. Mohammadi, and D. Salimi-Majd. *Investigation of nonlinear post-buckling delamination in curved laminated composite panels via cohesive zone model.* Thin-Walled Structures, 154:106797, 2020.

[12] **M. Moradi** and M. S. Safizadeh. *Experimental and numerical simulation of thermography for detection of disbond in repaired aluminum by composite patch.* In Persian, Journal of Science and Technology of Composites, 6(4):591–600, 2020.

[13] **M. Moradi** and M. S. Safizadeh. *Experimental and numerical study of the effect of using polyurethane instead of teflon strip to simulate debonding defect in composite patch repairs aluminum plate under thermography inspection.* Composites Part B: Engineering, 175:107176, 2019.

[14] **M. Moradi** and M. S. Safizadeh. *Detection of edge debonding in composite patch using novel post processing method of thermography.* NDT & E International, 107:102153, 2019.

[15] **M. Moradi**, B. Ameri, and B. Mohammadi. *Post buckling behavior analysis of unidirectional saddle shaped composite panels containing delaminations using cohesive zone modeling.* In Persian, Journal of Science and Technology of Composites, 6(2):255–264, 2019.

[16] **M. Moradi** and M. S. Safizadeh. *Edge disbond detection of carbon/epoxy repair patch on aluminum using thermography.* Composites Science and Technology, 179:41–53, 2019.

[17] M. Bayat, M. Safizadeh, and **M. Moradi**. *Numerical and experimental study for assessing stress in carbon epoxy composites using thermography.* Infrared Physics & Technology, 98:108–113, 2019.

Conference papers

[1] **M. Moradi**, F.C. Gul, J. Chiachío, R. Benedictus, and D. Zarouchas. *Advanced Health Monitoring of Composite Structures Through Deep Learning-Based Analysis of Lamb Wave Data for Developing Health Indicators.* In International Workshop on Structural Health Monitoring, volume 14, pages 1209-1216. DEStech publications, Inc., 2023 (Stanford, USA).

[2] F.C. Gul, **M. Moradi**, R. Benedictus, R. Hadjria, Y. Lugovtsova, and D. Zarouchas. *Delamination Size Prediction for Compressive Fatigue Loaded Composite Structures Via Ultrasonic Guided Wave Based Structural Health Monitoring.* In International Workshop on Structural Health Monitoring, volume 14, pages 1217-1224. DEStech publications, Inc., 2023 (Stanford, USA).

[3] **M. Moradi**, J. Chiachío, and D. Zarouchas. *Developing health indicators for composite structures based on a two-stage semi-supervised machine learning model using acoustic emission data.* In SMART, volume 10, pages 923–924. ECCOMAS, 2023 (Patras, Greece).

[4] S. Galiana, **M. Moradi**, P. Wierach, and D. Zarouchas. *Acousto-ultrasonic composite transducers integration into thermoplastic composite structures via ultrasonic welding.* In SMART, volume 10, pages 956–965. ECCOMAS, 2023 (Patras, Greece).

[5] **M. Moradi**, P. Komninos, R. Benedictus, and D. Zarouchas. *Interpretable neural network with limited weights for constructing simple and explainable HI using SHM data.* In Annual Conference of the PHM Society, volume 14, 2022 (Nashville, USA).

[6] **M. Moradi**, A. Broer, J. Chiachío, R. Benedictus, and D. Zarouchas. *Intelligent health indicators based on semi-supervised learning utilizing acoustic emission data.* In European Workshop on Structural Health Monitoring, pages 419–428. Springer, 2022 (Palermo, Italy).

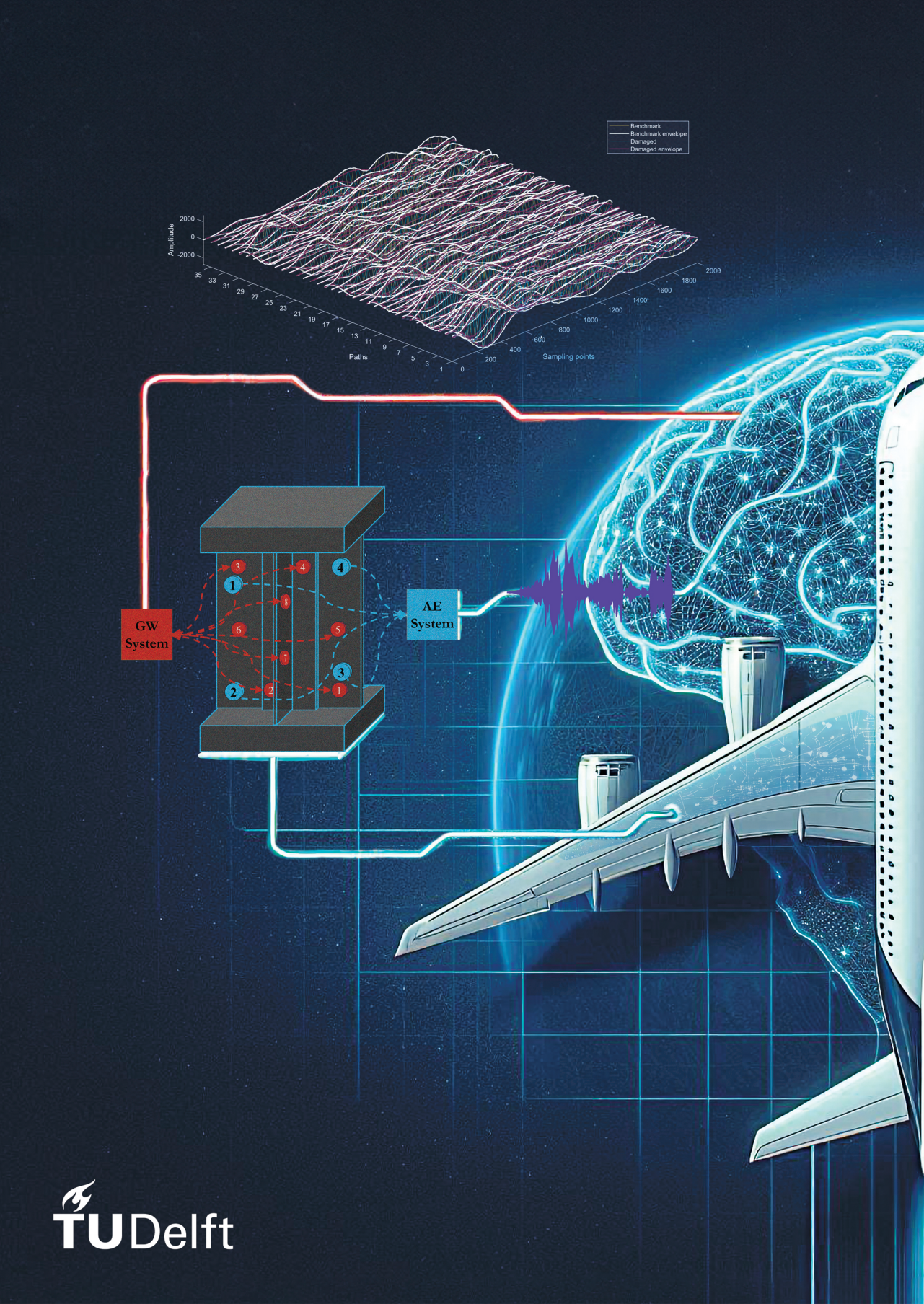
[7] **M. Moradi**, R. Ghorbani, S. Sfarra, D. Tax, and D. Zarouchas. *Infrared thermal defect identification and reconstruction of artworks using a spatiotemporal deep neural network.* In International Conference on Inverse Problems in Engineering, pages 503–510, 2022 (Chieti, Italy).

[8] **M. Moradi** and M. S. Safizadeh. *Numerical and experimental simulation of thermography for detection of debonding in repaired aluminum by carbon/epoxy patch.* In Iranian International NDT Conference, volume 5, 2019 (Tehran, Iran).

Book chapters

[1] **M. Moradi**, M. Kersemans, S. Sfarra, and D. Zarouchas. *Impact damage assessment in composite laminates using infrared thermography.* In *Non-destructive Testing of Impact Damage in Fiber-reinforced Polymer Composites: Fundamentals and Applications*, pages 215–250. Elsevier, 2024.

[2] R. Shrestha, **M. Moradi**, S. Sfarra, and W. Kim. *Infrared thermography: philosophy, approaches, analysis—processing, and guidelines.* In *Non-Destructive Material Characterization Methods*, pages 249–279. Elsevier, 2024.


Benchmark
Benchmark envelope
Damaged
Damaged envelope
Amplitude
2000
0
-2000
Paths
35
33
31
29
27
25
23
21
19
17
15
13
11
9
7
5
3
1
0
200
400
600
800
1000
1200
1400
1600
1800
2000
Sampling points
GW System
AE System
3
1
4
4
8
6
5
7
3
2
2
1

TU Delft