White Rose Research Online

eprints@whiterose.ac.uk
https://eprints.whiterose.ac.uk
Universities of Leeds, Sheffield and York

Deposited via The University of Sheffield.

White Rose Research Online URL for this paper:
https://eprints.whiterose.ac.uk/id/eprint/246807/

Version: Accepted Version

Article:
Worden, K., Brennan, D.J., Dardeno, T.A. et al. (2026) On the geometry of Population-Based Structural Health Monitoring: Part I — sheaves on graphs. Mechanical Systems and Signal Processing, 259. 114792. ISSN: 0888-3270

https://doi.org/10.1016/j.ymssp.2026.114792

© 2026 The Authors. Except as otherwise noted, this author-accepted version of a journal article published in Mechanical Systems and Signal Processing is made available via the University of Sheffield Research Publications and Copyright Policy under the terms of the Creative Commons Attribution 4.0 International License (CC-BY 4.0), which permits unrestricted use, distribution and reproduction in any medium, provided the original work is properly cited. To view a copy of this licence, visit http://creativecommons.org/licenses/by/4.0/

Reuse
This article is distributed under the terms of the Creative Commons Attribution (CC BY) licence. This licence allows you to distribute, remix, tweak, and build upon the work, even commercially, as long as you credit the authors for the original work. More information and the full terms of the licence here: https://creativecommons.org/licenses/

Takedown
If you consider content in White Rose Research Online to be in breach of UK law, please notify us by emailing eprints@whiterose.ac.uk including the URL of the record and the reason for the withdrawal request.

UNIVERSITY OF LEEDS
University of Sheffield
UNIVERSITY of York

On the Geometry of Population-Based Structural Health Monitoring: Part I – Sheaves on Graphs

K. Worden[1,*], D.J. Brennan, T.A. Dardeno, N. Dervilis, J. Poole, T.J. Rogers, G. Tsialiamanis, A.J. Hughes

Dynamics Research Group, School of Mechanical, Aerospace and Civil Engineering, University of Sheffield, Mappin Street, Sheffield S1 3JD, UK

Abstract

Population-Based Structural Health Monitoring (PBSHM) has recently emerged as a means of using data from multiple structures to enhance diagnostic capability in any given individual structure. As part of a sequence of papers on the foundations of PBSHM, a geometric picture was proposed for the methodology. Although this picture provided some insight into PBSHM generally, it could not provide a rigorous basis for study as it was based on a fibre-bundle viewpoint that did not exactly match the 'geometry' of PBSHM. Assuming that a rigorous framework could provide valuable guidance into designing workable PBSHM systems, the objective of this paper (and a companion paper) is to provide a rigorous geometrical picture. The current paper achieves this by eschewing the fibre bundle as the main geometrical object of interest and replacing it with a *cellular sheaf*. It is shown how a population of structures, represented in abstract form, can be encapsulated in a cellular sheaf, together with the SHM feature data for the structures. Furthermore, a geometrical operation – a variant of local PCA alignment – is introduced which actually implements (linear) transfer learning between structures and naturally provides a generalisation of some previously-proposed algorithms. The methodology is validated on a population of simulated structures in the context of novelty detection. Further analysis of a multi-class SHM problem exposes some issues for PCA-alignment and statistic-alignment algorithms generally where rotations are included. As this paper possibly represents the first use of sheaves in engineering dynamics, substantial background material is included here and other aspects of sheaf technology are discussed.

Keywords: Population-Based Structural Health Monitoring (PBSHM), cellular sheaves, transfer learning, local PCA alignment, vector diffusion maps.

1. Introduction

Although data-based *Structural Health Monitoring* (SHM) [1] is arguably established now as a means of embedding health assessment in the asset management process for engineering structures, there remain a number of implementation issues. Because data-based SHM is based on machine-learning technologies, one of the main barriers to widescale industrial uptake has proved to be the problem of obtaining training data which characterise the various health and damage states that an SHM system might be

*Corresponding author
Email address: k.worden@sheffield.ac.uk (K. Worden)

called upon to diagnose. One proposed means of overcoming this problem is to go from the monitoring of individual structures to populations; in this case, the probability that data are available for structural damage states for *some* structure in the population is potentially greatly increased. If such data are available, *Population-Based SHM* (PBSHM) [2], uses Transfer Learning (TL) [3], to enable diagnostics on one structure, supported by data from another/others. While [2] provides a useful, if terse, overview, far more detailed discussions of the foundations of PBSHM can be found in [4, 5, 6, 7, 8].

As indicated above, transfer learning is built into the foundations of PBSHM [6]; in fact, it is a crucial ingredient. A significant problem in machine learning generally, is that of *generalisation*. This issue can occur even in the monitoring of a single structure, where environmental variations around a structure create a situation where the probability distributions for a training set may differ from those of a testing set and classification of the latter data will fail. The problem is exacerbated in PBSHM, where it is generally the case that data distributions may vary considerably between structures. A classifier trained on one structure will generally fail on another. Transfer learning was developed in the machine learning community in order to solve the problem of generalising over large distribution shifts [3, 9]. There are many variants in terms of TL algorithms, but one which has proved popular in the SHM/PBSHM context is *Domain Adaptation* (DA). By inferring a mapping to a domain–invariant *latent space*, DA adapts the distributions of the domains, allowing a classifier trained using labelled data in a *source* domain to generalise to a *target*. Even within DA, different classes of algorithm have been proposed, one group based on Reproducing Kernel Hilbert Spaces (RKHS) has proved quite powerful and has been applied successfully in damage detection [10] and damage localisation problems for SHM [11]. In recent years, excellent performance in DA has been achieved with deep neural network architectures, based on 'fine-tuning' approaches [12]. Recent work has investigated the application of deep TL in an SHM context; examples being a fine tuning approach to image-based crack detection [13], and a deep-DA for damage detection and quantification, using high-dimensional features with abundant data [14]. There is also now a developed literature on the application of deep-DA to perform fault diagnosis in machines, with earlier examples including [15, 16, 17, 18].

A key element of PBSHM is that it should be possible to compare structures and determine an objective measure of their *similarity*; this is important because it is generally the case that transfer learning works best (or at all), if the *source* and *target* problems are similar to each other. The means of addressing this problem in PBSHM is to create an *abstract representation of structures*, where individual structures of interest are represented as points in a metric space; the metric then serves to measure the 'distance' between structures and can be used in a proxy for their similarity. This abstract representation is created by deriving a so-called *irreducible-element* (IE) model of each structure, and this model is a physics-based representation [5]. There is then a natural way of converting the IE models to *attributed graphs* (AGs), which serve as the abstract representation. The space of AGs is a metric space, where various measures of similarity can be defined in a rigorous manner [5, 19].

The basic problem of data-based SHM generally is to measure some damage-sensitive *feature* data from the structure of interest and to infer from this a health state. Thus, in PBSHM, each structure in the population will have some data 'attached' to it. This idea suggests a geometrical picture of PBSHM as discussed in [7]. This picture was initially based on the idea of a *fibre bundle* from modern(ish) differential geometry [20, 21]. In many ways, the fibre-bundle model is very natural for PBSHM as it can assign a separate feature space to each structure in a population; this is illustrated in Figure 1, where the 'base space' – which represents the population – is the set of structures of interest, with each represented as an attributed graph derived from an irreducible element model, as discussed above; each structure S_i, is a single point in the base space. The base space itself is then a network or graph and data for a given structure are then considered to reside in the *fibre* F_i above the *point* S_i, which will usually be a vector space. Movement of data for transfer learning between data spaces (as in domain adaptation [6], for example), then corresponds to 'horizontal' movement between fibres [21], with movement within fibres considered to be 'vertical'.

The fibres F_i represent the set of all possible feature sets for the structure S_i. For example, if the features of interest for structure S_i are the first four natural frequencies, then F_i is isomorphic to $\mathbb{R}^4$. The possible health states of the structure need not 'fill'[1] the fibre F_i; however, there is a specific health state which one can define for all structures, which denotes the normal or undamaged condition. Without loss of generality, one can always define coordinates in the fibre (by translation if necessary), so that the normal-condition state is at the origin. Now, one essential ingredient of a fibre bundle is a projection π which maps any point in a fibre F_i (or in the total space) to its corresponding structure S_i. One can then define a *section* of the bundle as any map s, such that $s \circ \pi$ is the identity on some region of the base space. With coordinates in the fibres as defined above, one can then define the *zero section* as the map which carries a structure to the feature data for its normal condition. If the section s is defined on the entire base space, it is referred to as a *global section*. Interestingly, for the mathematical objects discussed later in this paper, the only global section guaranteed to exist is the zero section.

While the picture is suggestive and intuitive, it is of limited use for analysis as it stands, because it possesses little mathematical rigour. One major problem is that the base space of graphs is necessarily a point set with very little extra structure. Although the base space is a *metric space*, which is essential for PBSHM, it does not have a topology and the picture thus lacks the important property of *continuity*. Faced with this issue, if one wishes to preserve some geometrical interpretation of PBSHM, one is faced with two main alternatives:

1. Modify the geometrical object of interest – in this case a fibre bundle – to some other mathematical object, where the graph-like nature of the base space is not an obstacle to rigour.

[1] In fact, as natural frequencies are required to be positive, the feature set just mentioned should actually be $\mathbb{R}^4_+$; however, this is not a vector space.

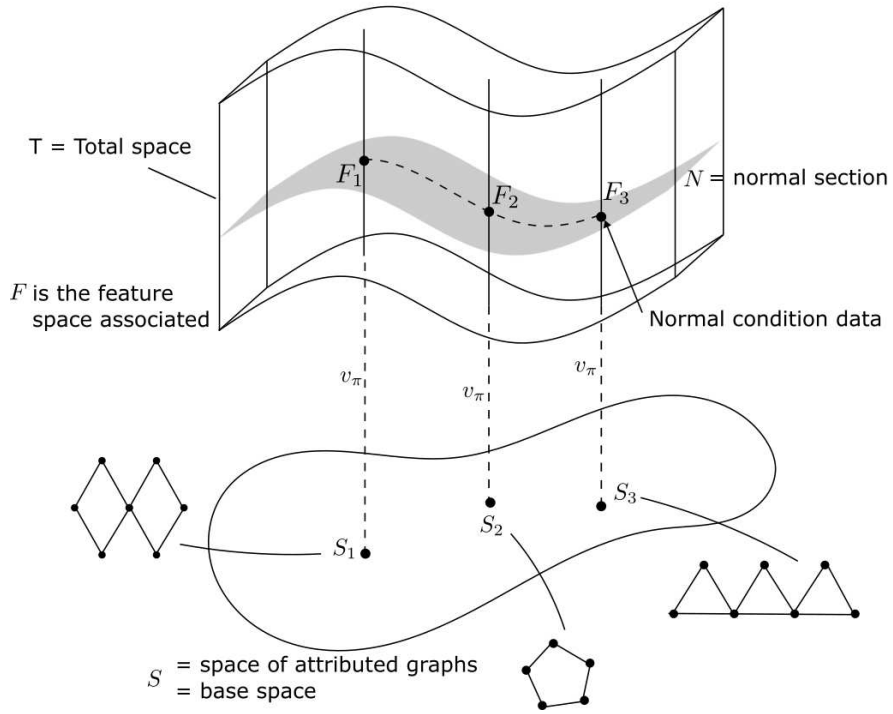


Figure 1: Schematic of fibre-bundle model of the PBSHM framework.

2. Modify the base space to allow some notion of continuity and thus of topology.

The current paper is the first of two, which will consider these alternatives: the second will be considered in the second paper in this short series, while the first will be addressed here. As part of the analysis here, a geometrical notion of movement between the 'fibres' will be established and thus a geometric interpretation of transfer. The idea here will be to present a theoretical framework for PBSHM which is backward-compatible with the studies so far and forward compatible so that it can guide development of the existing database formulation [2, 8].

Another point on which the fibre-bundle picture breaks down is as follows. For a true fibre bundle, *all* fibres must be isomorphic to a model fibre, say F. Now, for PBSHM, although the assumption is that each structure has some feature data, it may be that the data for different structures are different. For example, one might have measured the first four natural frequencies as features for structure S_1, while the features for S_2 are a set of 32 lines from a frequency-response function (FRF). In the first case, one has that $F_1 \simeq \mathbb{R}^4$, while in the second, one has that $F_2 \simeq \mathbb{C}^{32}$. The more general mathematical object discussed in this paper will actually allow this degree of flexibility if needed.

It will transpire that the required mathematical object is a *sheaf*. The objective here is to attach data to graphs and networks in a rigorous fashion, and as Hansen remarks in [22]; 'Sheaves are the canonical mathematical structure for attaching data to spaces'.

Furthermore, it is fact of algebraic geometry [23] that one class of sheaves – quasi-coherent sheaves – are equivalent to vector bundles. In fact, because the space of interest here is a space of graphs, rather a specialised type of sheaf is needed here – a *cellular sheaf*.

As much of the material on graphs and sheaves is likely to unfamiliar to practitioners of SHM, the paper contains a substantial tutorial component.

The layout of the paper is as follows: Section 2 gives a brief description of 'classical' sheaves on topological spaces; then Section 3 gives the definition of sheaves on graphs – *cellular* sheaves. The latter section includes a tutorial introduction to some necessary concepts and notation from graph theory. Section 4 explains how transfer learning might be realised as maps between stalks of cellular sheaves defined on population networks. Section 5 shows how one can construct linear transfer maps using an analogy with the alignment algorithm of Singer and Wu [24]. In Section 6, the transfer methodology is demonstrated on a population of simulated structures. Section 7 provides some discussion on the results presented here and the paper then concludes.

2. Sheaves

Although cellular sheaves (or sheaves on graphs) are quite different objects to 'classical' sheaves, it is useful to summarise the properties of the original constructs. Both types of sheaves will be denoted here by the symbol $\mathcal{F}$. So, how does one define a classical sheaf? The discussion here follows [25].

The immediate prerequisites are a topological space X and a family of open sets $\mathcal{O} = \{U_i\}$ on X, which cover the space, i.e. $\bigcup_i U_i = X$. One can define a sheaf of many mathematical objects now, but the definitions are usually introduced for a sheaf of Abelian (commutative) groups. First, one assumes some set $\mathcal{A}$ of Abelian groups[2]. Now, one proceeds via an intermediate definition; a *presheaf* of Abelian groups on X is then defined as an assignment of an element of $\mathcal{A}$ to all the open sets of X; i.e. an element of $\mathcal{A}$ denoted by $\mathcal{F}(U_i)$ for all $U_i \in \mathcal{O}$, together with a function, the *restriction* map ρ_U^V, where $V \subseteq U$. In the usual notation, one can say that the restriction of the sheaf to the open set U, denoted by $\mathcal{F}|_U$ is defined by $\mathcal{F}(U) = \mathcal{F}|_U = \rho_X^U(X)$. The restriction map is required to have the properties,

$$\rho_U^U = \mathbb{I} \tag{1}$$

where $\mathbb{I}$ is the identity map on, in this case, the set of Abelian groups $\mathcal{A}$, and,

$$\rho_U^W = \rho_V^W \circ \rho_U^V \qquad \text{for } W \subseteq V \subseteq U \tag{2}$$

[2]If the definitions were given with maximum rigour, one would say the *category* of Abelian groups; in fact, the main objects of interest here will be vector spaces, which are Abelian groups under addition. One can define a sheaf of objects from any category; for example, in algebraic geometry [23], one commonly encounters sheaves of commutative rings.

where U, V and W are in $\mathcal{O}$. One can see why Hansen regards sheaves as the natural means of attaching data to spaces [22]. Note that the sheaf attaches an entire Abelian group $A \in \mathcal{A}$ to each open set. One can also specify an element $a \in A$ for each open set U; this is referred to as a *section* of $\mathcal{F}(U)$.

The sections of the sheaf are clearly local; one might wish to move to a *global section*. This move would clearly require consistency conditions on the overlaps of the open sets. A good example of this need is if one has a presheaf of continuous functions on X. In this case, a global section would be a continuous function on the whole of X; the sections on the open sets would need to glue together in an appropriate fashion.

A *sheaf* then, is defined as a presheaf with some desirable conditions on the overlap of open sets. In fact, only one extra condition is needed – the *sheaf axiom* [25]. This requires that, if one is given sections s_i of $\mathcal{F}(U_i)$ and s_j of $\mathcal{F}(U_j)$, the restrictions of s_i and s_j to $U_i \cap U_j$ *must be equal*.

One should see now, the resemblance to a fibre bundle; the difference being that assignments of 'data' are made to open sets rather than points (unless the discrete topology is used, which would be rather restrictive). To get something like a fibre requires a little more work. Suppose a point $x \in X$ has an open neighborhood U. Now, consider the disjoint union $\mathcal{U}_x$ of *all* the open sets that contain x. Now, consider two sets $V, W \in \mathcal{U}_x$, together with sections $s \in \mathcal{F}(V)$ and $t \in \mathcal{F}(W)$. One defines an equivalence relation $\sim$ on $\mathcal{U}_x$ such that, $s \sim t$ if there is open neighbourhood $S \subseteq V \cap W$ of x, on which the restrictions of s and t are equal, i.e.

$$\rho_V^S(s) = \rho_W^S(t) \tag{3}$$

Note this represents a much stronger condition than saying than the sections must agree at x. If one denotes the elements of the equivalence class by (V, s) (so that $(V, s) = (W, t)$ above), one can denote the elements of the equivalence class by s_x and the class itself by $\mathcal{F}_x$. This construction essentially establishes a map $\mathcal{F}(V) \longrightarrow \mathcal{F}_x$ which takes the section s to $s_x \in \mathcal{F}_x$. The set $\mathcal{F}_x$ is called the *stalk* of $\mathcal{F}$ at x. This construction is rather abstract, and feels like a great deal of work; however, it is needed in order to recover a property similar to that in a fibre bundle, where a section over an open set in the base manifold is fixed by the values of the section at all the points in that set. In the sheaf case, a section s over U is uniquely determined by the values of the s_x for all points $x \in U$. This entails that a *zero section* over U follows from $s_x = 0$ for all $x \in U$.

3. Cellular Sheaves

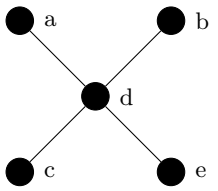
3.1. Graphs

As cellular sheaves are essentially sheaves on graphs, it will be necessary to establish some basic (and not so basic) structure and notation for graphs.

A graph G is defined by a set of nodes or vertices V and edges or links E, and so $G = (V, E)$ [26]. The edge and node set of a graph can alternatively be represented as $E(G)$ and $V(G)$, respectively. This notation is useful in contexts where it is necessary to define the vertex and edge sets for multiple graphs. In general here, the nodes of a graph will be denoted by $\{v_i; i = 1, \ldots, N_v\}$, where N_v is the number of vertices; edges will be denoted by symbols e_{ij} in general, if an edge exists between nodes v_i and v_j; the total number of edges in the graph will be denoted by N_e. For small graphs, individual labels may be used from the alphabet or integers. Edges between nodes v_i and v_j can also be denoted by $(v_i, v_j) \in E(G)$.

An *attributed graph* is obtained when the graph structure encodes additional information beyond its topology. In this case, nodes and/or edges can each have sets of parameters associated with them. It will shortly become clear that cellular sheaves allow the definition of attributed graphs.

It is also possible to define a graph by its adjacency list; this contains each node in V along with its neighbourhood, where the neighbourhood $N(v)$ for each node is defined as $N(v) = \{u \in V \mid (u, v) \in E\}$. An example of a graph and the corresponding adjacency list is shown in Figure 2. Neighbourhoods can be defined in more general theory as *open* or *closed*[3]; however, this could be confusing here as the open sets are not used specifically to define a topology, so these conventions will not be used.




```
graph =  { a :  [d]
           b :  [d]
           c :  [d]
           d :  [a, b, c, e]
           e :  [d]  }
```

Figure 2: A graph and its corresponding adjacency list. In this particular case, the adjacency list has been represented using a Python dictionary.

Graphs are also conveniently represented by their *adjacency matrix* $A(G)$, which sets the element of the matrix a_{ij} to be unity if the edge e_{ij} exists, and to zero otherwise. The adjacency matrix for the graph in Figure 2 is given by,

$$A(G) = \begin{pmatrix} 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 & 0 \\ 1 & 1 & 1 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 \end{pmatrix} \tag{4}$$

[3]The neighbourhood N as usually defined, does not contain the node itself; this is known as the *open neighbourhood* of v. The *closed neighbourhood* of a node v, contains the node itself and is defined as $N[u] = N(v) \cup v$.

where 'self-edges' (loops) are ignored. It will be clear that the adjacency matrix will always be symmetric for undirected graphs.

A *path* between two nodes v_i and v_j is any sequence of nodes pairwise connected by edges, starting at v_i and ending at v_j. If there is a path from any node in G to every other node in G, then the graph is *connected*. Because of the way that structure graphs are defined in PBSHM, they will always be connected, otherwise the structure will fall into two disjoint substructures. Figure 3 shows two 'cartoon' structure graphs from [5]; the first is from an IE model of a wind turbine, the second is for an aeroplane.

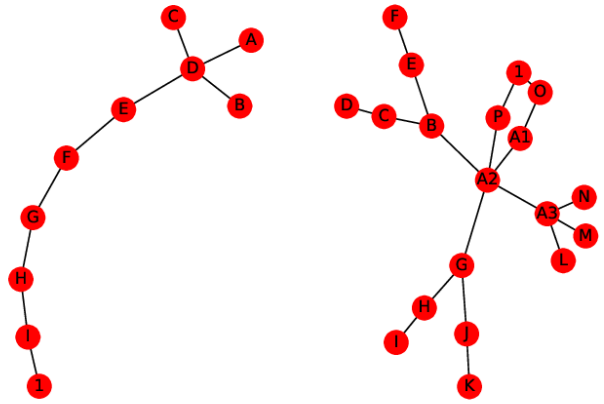


Figure 3: Two very simple structure graphs from [5]; the first represents a wind turbine, while the second represents an aeroplane.

It will prove convenient to introduce a *boundary operator* ∂ for graphs; this is defined such that $\partial e = \{v_1, v_2\}$, where the vertices v_1 and v_2 bound the edge e. Note that in complete generality, two vertices might have multiple edges.

An important idea in general graph theory is that graphs can be *undirected* or *directed*. The graphs presented so far are all examples of undirected graphs. In a directed graph, each edge is given a *direction*, which is essentially an arrow pointing from one boundary vertex to the other. This makes the boundaries of edges into ordered pairs. An *orientation* for a graph is a choice of direction for its edges. If a sequence of edges leads back to the starting vertex and follows the directions of the edges, it is referred to as a *cycle*. As an example, Figure 4 shows a three-node directed graph with an orientation that makes it into a cycle.

It is clear that the unsigned adjacency matrix is not sufficient to uniquely specify a directed graph $G(E, V)$; instead, one can define the *incidence matrix* $B(G)$, which is specified as follows. Suppose nodes v_i and v_j are joined by an edge e_k. In this case, one sets $B_{ij} = [v_i; e_k]$, where $[v_i; e_j] = -1$ if the orientation of edge e_k has its arrow

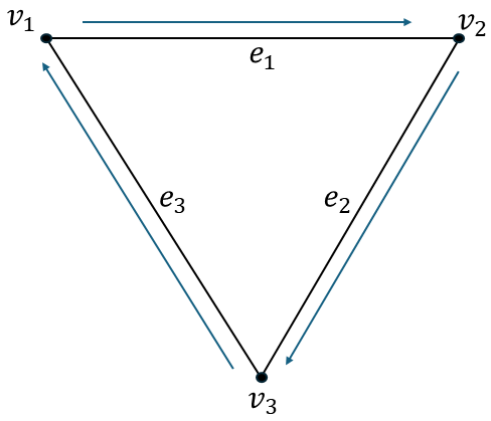


Figure 4: Three-node directed graph which is a cycle.

pointing *away* from v_i, and $[v_i; e_k] = +1$ if the arrow for e_k points *towards* v_i. If v_i does not bound e_k, then $[v_i; e_k] = 0$. Note that the matrix $B(G)$ will thus be of dimensions $|V(G)| \times |E(G)|$ where $|S|$ denotes the number of elements in the set S. The incidence matrix for the graph in Figure 4 will thus be,

$$B(G) = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \end{array} \begin{array}{c} \begin{array}{ccc} e_1 & e_2 & e_3 \end{array} \\ \begin{pmatrix} -1 & 0 & 1 \\ 1 & -1 & 0 \\ 0 & 1 & -1 \end{pmatrix} \end{array} \tag{5}$$

One can include directions in the adjacency matrix by adopting the conventions that $A_{ij} = -1$ if v_i is directed towards v_j along an edge, $A_{ij} = 1$ if v_i is directed towards v_j, and $A_{ij} = 0$ if vertices v_i and v_j are not connected by an edge. The adjacency matrix for the underlying undirected graph is obtained by simply taking the modulus of the elements. The (signed) adjacency matrix for the triangle graph is thus,

$$A(G) = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \end{array} \begin{array}{c} \begin{array}{ccc} v_1 & v_2 & v_3 \end{array} \\ \begin{pmatrix} 0 & -1 & 1 \\ 1 & 0 & -1 \\ -1 & 1 & 0 \end{pmatrix} \end{array} \tag{6}$$

and such matrices will always be antisymmetric.

Another important matrix defined for a graph is the *degree matrix* $D(G)$; this is a diagonal matrix of dimensions $|V(G)| \times |V(G)|$ where D_{ii} equals the number of edges meeting v_i. For the undirected triangle graph, one has,

$$D(G) = \begin{array}{c} \\ v_1 \\ v_2 \\ v_3 \end{array} \begin{array}{c} \begin{array}{ccc} v_1 & v_2 & v_3 \end{array} \\ \begin{pmatrix} 2 & 0 & 0 \\ 0 & 2 & 0 \\ 0 & 0 & 2 \end{pmatrix} \end{array} \tag{7}$$

and for directed graphs one can count the *indegree* or *outdegree*. Clearly, it is the case that the trace of the degree matrix satisfies,

$$\mathrm{Tr}(D(G)) = 2|E(G)| \tag{8}$$

The *Laplacian* $L(G)$ of a graph G is defined as,

$$L(G) = D(G) - A(G) \tag{9}$$

It may not be immediately obvious, but the graph Laplacian can be thought of as a matrix form of the negative discrete Laplacian on a graph which approximates the negative continuous-space Laplacian by finite differences. Clearly, $L(G)$ has dimensions $|V(G)| \times |V(G)|$. In order to see this, one has to introduce the idea of *functions* on the graphs. The most useful case one can deal with is that of real-valued function f on a graph G; this is simply an assignment of a real-value to each vertex of the graph, so $f : V(G) \longrightarrow \mathbb{R}$. Such a function is specified by its values on the set of vertices; i.e. $f = (f(v_1), f(v_2), \ldots, f(v_n))^T$ where $n = |V(G)|$, or $(f(1), f(2), \ldots, f(n))$ in a simplified notation. As this is a vector of dimension $V(G)$, one can multiply it by the transpose of the incidence matrix B^T, which has dimensions $|E(G)| \times |V(G)|$. To make things concrete, one can consider the directed triangle graph again, so that, for a function f defined on the graph, one has,

$$B(G)^T f = \begin{pmatrix} -1 & 1 & 0 \\ 0 & -1 & 1 \\ 1 & 0 & -1 \end{pmatrix} \begin{pmatrix} f(1) \\ f(2) \\ f(3) \end{pmatrix} = \begin{pmatrix} f(2) - f(1) \\ f(3) - f(2) \\ f(1) - f(3) \end{pmatrix} \tag{10}$$

Now, note that this vector has dimension $|E(G)|$ and so has one element per edge. Because of the ordering chosen for the edges, one has that the values of the product associated with the edges e_1, e_2 and e_3 are $f(2) - f(1)$, $f(3) - f(2)$ and $f(1) - f(3)$ respectively; these are the *discrete differences* of the function f across the edges, taking account of orientation. Thus B^T has an interpretation as a discrete difference operator $\nabla = B^T$ on functions on the graph G. Given the definition of $B_{ij} = [v_i; e_j]$ from earlier, it is not too difficult to see that this result is true generally. Now, given the dimensions of $B^T f$, it is clear that the product $B(B^T f) = (BB^T)f$ is defined; for the triangle graph, one finds that,

$$(BB^T)f = -\begin{pmatrix} f(2) - 2f(1) + f(3) \\ f(3) - 2f(2) + f(1) \\ f(1) - 2f(3) + f(2) \end{pmatrix} \tag{11}$$

Now, this vector is of dimension $|V(G)|$ and, because of the ordering of vertices again, the elements are associated with v_1, v_2 and v_3. Just looking at the v_1 element, one sees that this is $-(f(2) - 2f(1) + f(3))$, which is the (negative) second discrete derivative of f at the vertex v_1, respecting the orientation of the edges meeting v_1. The same is true of the other components, and in fact is true for general graphs again, although the analysis will not be presented here. The product is thus the (negative) Laplacian,

$$L(G) = B(G)B(G)^T \tag{12}$$

It takes more work to show that this agrees with the definition in equation (9); however, the argument will not be presented here, as (12) will prove of more direct relevance for cellular sheaves. The curious reader can consult [27], which is a comprehensive reference on modern graph theory.

The question remains as to how to specify the metric on the space of AGs; one way to do this is via the *maximum common subgraph* (MCS). Two graphs are clearly more similar if they have more common structure, and this is captured in the size of their MCS. Furthermore, the MCS makes sense from a PBSHM point of view; common subgraphs in the AGs correspond to common substructures in the structures of interest. If an SHM problem is to be transferred between structures, the process is clearly more likely to succeed if the problem occurs within a common substructure. The process of extracting the MCS between two graphs is computationally demanding; the basic graph-theoretic algorithm used in [5] is the *Bron-Kerbosch* algorithm [28]. Once the MCS is established, this can be converted into a metric distance which is smaller when two structures share larger MCSs; the metric used in here and described in [5], is based on the *Jaccard similarity coefficient* or *Jaccard index* [29].

Given two graphs G and H, the Jaccard index is based on the size of the maximum common subgraph G', thus providing a measure of the similarity based purely on the topology of the structures, the coefficient/index J_v is defined by,

$$J_v(G, H) = \frac{| V(G') |}{| V(G) | + | V(H) | - | V(G') |} \tag{13}$$

where $|V(G)|$ is the number of nodes/vertices in G etc. The index is bounded between 0 and 1.

The Jaccard index can be used to generate the Jaccard distance $D_v(G, H) = 1 - J_v(G, H)$; this distance is a metric on the set/space of structures/attributed graphs. As an illustration, Figure 5 shows the maximum common subgraph between the two

structures in Figure 3, The subgraph corresponds to the path (and branches) running from A to H in the wind turbine graph and that running from M to F in the aeroplane graph; in fact there are multiple chains from M that have the same subgraph structure.

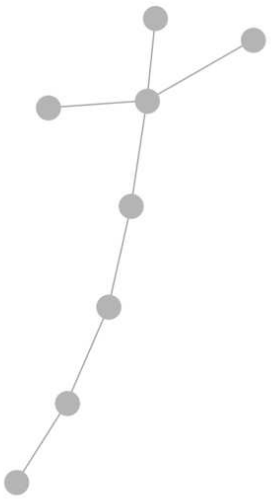


Figure 5: Maximum common subgraph between the two AG models illustrated in Figure 3.

In this case, consulting Figures 3 and 5, one has $|V(G)| = 10$, for G the 'turbine' graph, $|V(H)| = 19$ for H the 'aeroplane' graph, and $|V(G')| = 8$ for their MCS; thus by equation (13), the Jaccard index is $8/21$ and the Jaccard distance is $13/21$.

A number of other distance metrics are discussed in [19, 30].

3.2. Cellular Sheaves: Basic Principles

The discussion here follows that of [22], albeit with some modified notation; it starts with the general definition.

Definition 1: Let $G = G(E, V)$ be a graph. A *(cellular) sheaf* on G consists of a vector space $\mathcal{F}(v)$ for every vertex $v \in V(G)$, a vector space $\mathcal{F}(e)$ for each edge, and a linear transformation $\mathcal{F}(\rho)_{v \in \partial e}$ for each incident vertex-edge pair; i.e. for all $v \in \partial e$.

Note that, on this basis, each $\mathcal{F}(e)$ contains the images of two inward-directed maps from the spaces on its boundary nodes. In general, the analysis here will allow the maps to be *affine transformations*.

The spaces $\mathcal{F}(v)$ and $\mathcal{F}(e)$ are called the *stalks* over the vertices and edges respectively. The terminology mirrors that of a classical sheaf; however, the stalks here are restricted to be vector spaces, rather than objects in a general category.

The map between node v and edge e will be denoted ρ_v^e; these maps are termed *restriction maps*. Although the terminology mirrors that of a classical sheaf, it does not carry any

parallel meaning because the notion of an open set is absent here. Because the restrictions are affine, they will generally be invertible; this means that one can compose maps and define $\tau_{v_1}^{v_2} = (\rho_{v_2}^e)^{-1} \circ \rho_{v_1}^e$. clearly these maps are from $\mathcal{F}(v_1)$ to $\mathcal{F}(v_2)$; i.e. between nodes that bound the edge e. These maps will be referred to as *transfer maps*, for reasons that will shortly become clear.

The application to PBSHM should be clear; the stalks will represent the feature spaces for data-based SHM. In this case, the vector spaces will all be of the form $\mathbb{R}^n$ – e.g. the first n natural frequencies of a structure – or $\mathbb{C}^m$ – e.g. m spectral lines from an FRF. There are two distinct situations in PBSHM where these sheafs might apply, depending on the interpretation of the relevant graph G:

1. The graph G is an IE model of an individual structure. In this case, one can think of a set of sensors $S = \{s_1, s_2, \dots, s_{n_S}\}$ which are being used for monitoring the structure. This set can be assumed to be non-empty, because there is no data-based SHM if otherwise. Now, as the sensors must have a physical location within some element; their data are associated with that element, which becomes a node v in the graph representation. If there are multiple sensors on the element, one can take the direct sum of their feature spaces to create an overall feature space associated with the node; this in turn becomes the space $\mathcal{F}(v)$.

   There is a subtlety here. While the raw data of a given sensor can be assigned a location on the structure, the derived features may be *global*; for example, natural frequencies. Other features may be local; e.g., the components of a modeshape or spectral lines from an FRF. Different types of features may also combined in a direct sum as described above and one might thus arrive at mixtures of local and global features. At the cost of a little redundancy, it will be assumed that copies of any global features are assigned locally. While this approach is perhaps not the most elegant, it does leave the basic form of the cellular sheaf intact; an alternative would be define a 'global stalk' for the whole graph, and this could be considered contrary to the view that stalks are themselves local objects.

   In this situation, the $\mathcal{F}(u)$ for all nodes u without sensors will be empty of data (as will all the $\mathcal{F}(e)$ as a rule; however, one can at least conceive of a situation where one might wish to monitor an interface or joint). Suppose one wished to *infer* data from a position with no sensor, the transfer maps could potentially be used for such inferential measurement by using transfer from a neighbouring node for which there *are* data. A potential means of addressing such problems via types of neural networks will be discussed later.
2. The graph G is the network of structure graphs that represent the PBSHM population. Then G is itself a set of graphs corresponding to structures. Suppose that node $v_1 \in G$ represents a structure graph S_1 and node v_2 represents S_2. In this case, the space $\mathcal{F}(v_1)$ (resp. $\mathcal{F}(v_2)$) represents the feature space for S_1 (resp. S_2). The map $\tau_{v_1}^{v_2}$, then carries feature data from S_1 to S_2, which is the objective of transfer learning [6].

   The connectivity of the graph G is of great interest here. In principle, one could

allow transfer between any two structures S and S', and then G would be fully connected. This is a rather uninteresting situation and most pairings of structures would result in negative transfer. One of the ideas of PBSHM is that transfer should only be attempted if there is a reasonable chance of success. It is hypothesised that transfer will most likely be successful if the structures (or their feature data), are sufficiently similar in some sense. This requirement is the reason why the structure space should be a metric space. Suppose that one could define a threshold d_{max} such that a metric distance $d(S, S') < d_{max}$ ensured positive transfer; in this case, edges would only connect nodes (structures) that satisfy the distance condition. The computation of such thresholds is a very open research problem in PBSHM, although some interesting preliminary work can be found in [31].

The idea of using affine transformations for transfer has been given a statistical motivation in [32]; the idea of using cellular sheaves in this paper is to provide a *geometrical* motivation.

3.3. Sections

As in the case of classical sheaves, one can define sections for cellular sheaves; in the context of PBSHM, a section will be an assignment of features to each of the structures (where data are available). Sections are always definable locally; however, consistency conditions are required if the local sections are to glue together in order to form a global section. These conditions are dependent on the restriction maps, which are themselves local to edge-vertex pairs.

Definition 2: Let $\mathcal{F}$ be a (cellular) sheaf on a graph $G(E, V)$ and let $W \subset V(G)$ be a subset of the vertices of G. A *section* [22] on G is a choice of vectors $\underline{x}_v \in \mathcal{F}(v)$ for each $v \in W$, such that whenever $\partial e = \{v_1, v_2\}$, one has,

$$\rho^e_{v_1}(\underline{x}_{v_1}) = \rho^e_{v_2}(\underline{x}_{v_2}) \tag{14}$$

This condition means that, if one follows any sequence of vertices $v_1, v_2, \ldots, v_n, v_1$ with the $v_i \in V(W)$ and consecutive v_1 connected by an edge, one will arrive at the same $\underline{x}_1$ as that in the initial stalk.

If $W = V(G)$, the conditions (14) define a *global section*; all the local information in the restriction maps glues together to give a global picture.

Consider the triangle graph again; Figure 6 shows a corresponding sheaf with stalks and restriction maps indicated; the underlying undirected graph is shown for simplicity. To avoid confusion in general, the arrows for orientations on graph figures will be shown in blue, while the arrows for restriction maps will be shown in green. To make more direct contact with the fibre bundle picture in Figure 1, Figure 7 shows the cellular sheaf structure of Figure 6 with the stalks show as vertical spaces. At the risk of a little repetition, the stalks will be feature spaces for SHM or PBSHM and can be real vector

spaces (e.g., for natural frequencies) or complex (e.g., spectral lines). As discussed above; in the first interpretation of the sheaf, over a single structure, the restriction maps can represent inferential measurements, while in the second case, over a population, the maps can represent transfer.

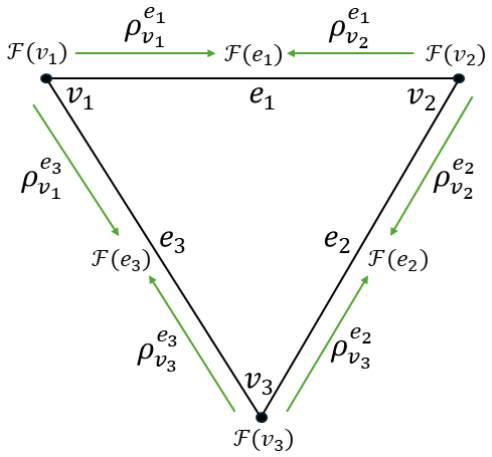


Figure 6: Undirected triangle graph showing stalks of a cellular sheaf together with its associated restriction maps.

If one follows the sequence v_1, v_2, v_3, v_1, the map from $\mathcal{F}(v_1) \longrightarrow \mathcal{F}(v_1)$ composes as,

$$(\rho_{v_1}^{e_3})^{-1} \circ \rho_{v_3}^{e_3} \circ (\rho_{v_3}^{e_2})^{-1} \circ \rho_{v_2}^{e_2} \circ (\rho_{v_2}^{e_1})^{-1} \circ \rho_{v_1}^{e_1} \tag{15}$$

and if one imposes the consistency conditions, the composition collapse to the identity as required to give a global section.

As the stalks are vector spaces, there is a natural way to add sections pointwise in each stalk. The restriction maps are linear, so the sum of sections is a section and the sections over any $W \subset V(G)$ form a vector space; this is denoted $\Gamma(W, \mathcal{F})$. For a classical sheaf, this space represents the zeroth sheaf cohomology group on $\mathcal{F}$ and is also denoted $H^0(W, \mathcal{F})$ [23].

Definition 3: Let $G(V, E)$ be a graph and F be a vector space; the *constant sheaf* $\mathcal{F}_F$ has stalks $\mathcal{F}(v) = F$ for all $v \in V(G)$ and $\mathcal{F}(e) = F$ for all $e \in E(G)$. (One can think of F as the analogue of the standard fibre in a bundle.) Furthermore, all the restriction maps ρ_v^e are the identity. For any $W \subset V(G)$, $\Gamma(W, \mathcal{F}_F)$ consists of all *locally constant* functions on W with values in F.

Nontrivial global sections are special; however, every sheaf has one global section, which is the constant section which is zero everywhere; this is the *zero section*. The zero section is of some importance in PBSHM. Suppose one shifts all feature data for each structure so that the mean (centroid) is at the origin of the feature space, then the normal condition

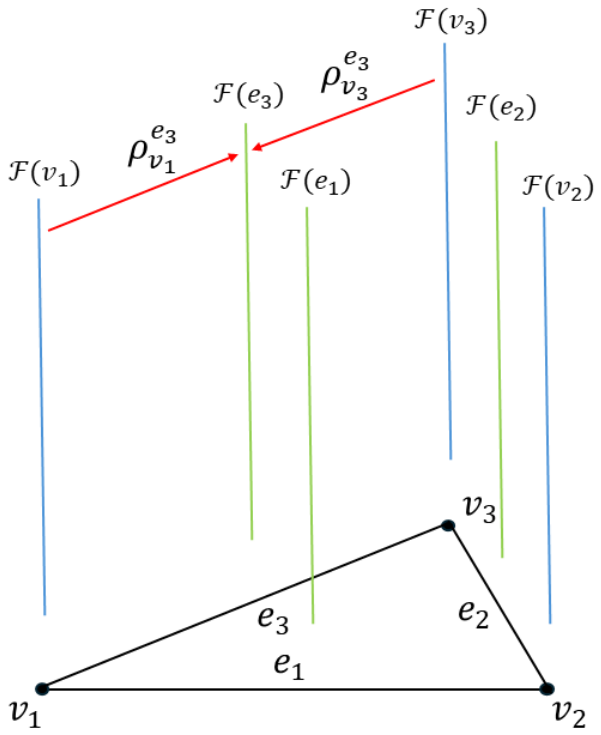


Figure 7: Undirected triangle graph showing stalks of a cellular sheaf together with associated restriction maps. In this case, the stalks are plotted as vertical lines to make evident the analogy with the fibres in Figure 1. For simplicity, only the restriction maps over edge e_3 are shown.

across the population is defined by the zero section on the PBSHM sheaf. This condition means that normal condition is defined globally.

One can illustrate these ideas with a nice example from [22]. Consider the sheaf over the triangle graph shown in Figure 8. This is very nearly the constant sheaf; all the stalks are isomorphic to $\mathbb{R}$ and all the restriction maps are the identity, except for $\rho_{v_1}^{e_1} = -1$. Suppose that this sheaf has a global section and so the compatibility conditions hold. Further suppose that the value of the section in $\mathcal{F}(v_i)$ is given by x_i. The compatibility condition for edge e_2 asserts that $x_3 = x_2$ and compatibility for e_3 that $x_1 = x_3$, so it follows that $x_1 = x_2$. However, compatibility for edge e_1 says that $x_1 = -x_2$ – a contradiction. The compatibility conditions do not hold globally unless $x_1 = x_2 = x_3 = 0$, showing that the only global section is the zero section.

3.4. Cellular Sheaf Cohomology

Mathematics is often concerned with classifying its objects into meaningful subsets. In the case of topological spaces, the main problem is to classify them up to *homeomorphism*; this means that, if two spaces are related by a continuous map, who's inverse is also continuous, they are considered to be the same. This is a very difficult and generally

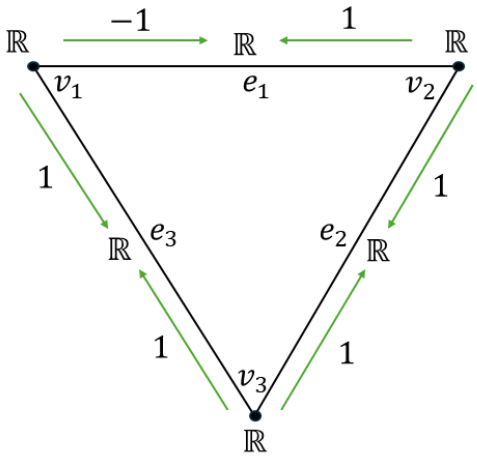


Figure 8: Sheaf over the triangle graph where the only global section is the zero section.

unsolved problem. One way of approaching the problem is by computing *topological invariants* for spaces; if the invariants are different, then the spaces are not homeomorphic. The simplest example of a topological invariant is the dimension of a space. Topological invariants can have more interesting structure than simple integers; importantly, they can be *groups*. If two spaces are homeomorphic, then they will have the same groups as invariants. Computing such groups is the remit of the subject of *algebraic topology* [33]; the main theories are homotopy, homology and cohomology. In some ways, the groups are related and the most convenient choice will be dictated by the problem.

In classical sheaf theory, cohomology provides a powerful means of characterising the topology of sheaves. There is a direct analogy for cellular sheaves; however, the theory is simplified by the fact that there are only two cohomology groups for sheaves over graphs – H^0 and H^1.

Definition 4: Let $\mathcal{F}$ be a cellular sheaf over a graph $G(V, E)$. There are two spaces of *cochains* of $\mathcal{F}$. A zero-dimensional cochain (or 0-cochain) is a choice of vector $\underline{x}_v \in \mathcal{F}$ for all $v \in V(G)$. A one-dimensional cochain is a choice of vector $\underline{x}_e \in \mathcal{F}(e)$ for all $e \in E(G)$.

Clearly, addition of cochains is defined. The space of 0-cochains is denoted by $C^0(G, \mathcal{F})$ and the space of 1-cochains by $C^1(G, \mathcal{F})$. It is clear that,

$$C^0(G, \mathcal{F}) = \bigoplus_{v \in V(G)} \mathcal{F}(v) \tag{16}$$

$$C^1(G, \mathcal{F}) = \bigoplus_{e \in E(G)} \mathcal{F}(e) \tag{17}$$

There is no requirement of consistency for the choices of the $\underline{x}_v$ in the space of 0-cochains, so in this case $\Gamma(G, \mathcal{F}) \subset C^0(G, \mathcal{F})$. In fact, the $\Gamma(G, \mathcal{F})$ is the kernel of a linear map $\delta : C^0(G, \mathcal{F}) \longrightarrow C^1(G, \mathcal{F})$ – the *coboundary* map – which is defined as follows.

First, define an orientation for G, by assigning a direction to each $e \in E(G)$; i.e. an ordering for the pair $\{v_1, v_2\} = \partial e$. The incidence numbers $[v_i; e]$ are then defined as before.

The coboundary map $\delta : C^0(G, \mathcal{F}) \longrightarrow C^1(G, \mathcal{F})$ is then defined on the vertex stalks by,

$$\delta(\underline{x}_v) = \sum_{v \in \partial e} [v; e] \rho_v^e(\underline{x}_v) \tag{18}$$

Now, because δ is a linear combination of the ρ_v^e, it is clear that the image of δ is in $\mathcal{F}(e)$, so the action of the map can also be defined by its 'output' on each edge stalk,

$$(\delta x)_e = \rho_v^e(\underline{x}_v) - \rho_u^e(\underline{x}_u) \tag{19}$$

where the oriented $\partial e = (u, v)$ (note the change in parentheses).

The statement that $\Gamma(G, \mathcal{F}) = \ker \delta$ can now be explained. The kernel condition means that a 0-cochain $c \in \ker \delta$ satisfies $\delta c = 0$. Now, suppose that the vertices are labelled from 1 to n, then a 0-cochain c is an assignment of vectors $c = (\underline{x}_1, \ldots, \underline{x}_n)$ where $\underline{x}_i \in \mathcal{F}(v_i)$. Furthermore, assume that the edge orientations specify the ordered pairs (v_i, v_j) where $v_i \in \partial e$ for each e. Now, according to equation (19), one has $(\delta x)_e = \rho_{v_j}^e(\underline{x}_{v_j}) - \rho_{v_i}^e(\underline{x}_{v_i})$; however, if the consistency conditions hold, then $\rho_{v_j}^e(\underline{x}_{v_j}) = \rho_{v_i}^e(\underline{x}_{v_i})$ and thus $\delta c = 0$. Now, if the consistency conditions hold across each edge then c represents a global section and $\Gamma(G, \mathcal{F}) = \ker \delta$ follows.

In terminology parallel to the classical theory, one sees that $\Gamma(G, \mathcal{F})$ is thus the zeroth cohomology space $H^0(G, \mathcal{F})$. Now, if all the 0-cochains $c \in C^0(G, \mathcal{F})$ are global sections, then $\delta c = 0$ for all c and $H^0(G, \mathcal{F})$ is trivially zero; this provides the interpretation that $H^0(G, \mathcal{F})$ describes *obstructions* to defining global sections. If $c = (\underline{0}, \ldots, \underline{0})$, then $\delta c = 0$ and it is clear that the zero section is always global.

The *cokernel* of δ is defined as the quotient space $C^1(G, \mathcal{F})/\text{Im } \delta$ and is the first cohomology space $H^1(G, \mathcal{F})$ of $\mathcal{F}$.

Consider the constant sheaf on a graph G, where $F = \mathbb{R}$. In this case, one has that the elements of $C^0(G, \mathcal{F})$ are real vectors of dimension $|V(G)|$ and the elements of $C^1(G, \mathcal{F})$ are real vectors of dimension $|E(G)|$. As δ is a linear operator, this suggests that it can be represented by an $|E(G))| \times |V(G)|$ matrix. Now, for a constant sheaf, one has that the only 0-cochain is $\underline{v} = (x, x, \ldots, x)$ of dimension $|V(G)|$ for some $x \in \mathbb{R}$, and the only 1-cochain is $\underline{e} = (x, x, \ldots, x)$ of dimension $|E(G)|$. Now, representing the coboundary operator by a matrix Δ, one has that $\underline{e} = \Delta \underline{v}$. Assume that the edges are ordered 1 to $|E(G)|$ and the vertices are ordered 1 to $|V(G)|$. As mentioned earlier, all the restriction maps on the constant sheaf are equal to the identity. From the matrix equation proposed, one has that $x_{e_i} = \sum_{j=1}^{|V(G)|} \Delta_{ij} x_{v_j}$. Now, equation (18) has that for edge e_i, one has $x_{e_i} = \sum_{v \in \partial e_i} [v; e_i] x_v$, as all the restriction maps are the identity. Supposing that $\partial e_i = (v_j, v_k)$ (directed), then $x_{e_i} = [v_k; e_i] x_{v_k} - [v_j; e_i] x_{v_j}$. Comparing

with the matrix operation by Δ, one sees that $\Delta_{ik} = [v_k; e_i]$ and $\Delta_{ij} = -[v_j; e_i]$. Finally, comparing with the definition of the incidence matrix – $B_{ij} = [v_i; e_j]$ – and allowing for the orientation of the edge e_i, which will impose $[v_j; e_i] = -[v_k; e_i]$ one sees that $\Delta_{ik} = B_{ki}$, so the matrix Δ is B^T, where B is the incidence matrix.

By an abuse of notation, one might say that,

$$\delta = B^T \tag{20}$$

in the highly-restricted case of the constant sheaf with $F = \mathbb{R}$. This motivates the idea that δ more widely is a significant generalisation of the incidence matrix (transposed) to sheaves (and furthermore recall that the incidence matrix can be thought of as a discrete differential operator).

Now, as $C^1(G, \mathcal{F})$ is a vector space, it is equipped with an inner product, so the product $< c_e, \delta c_v >$ is defined for cochains $c_v \in C^0(G, \mathcal{F})$ and $c_e \in C^1(G, \mathcal{F})$. One can use this structure to define an adjoint operator $\delta^* : C^1(G, \mathcal{F}) \longrightarrow C^0(G, \mathcal{F})$ by,

$$< c_e, \delta c_v > = < \delta^* c_e, c_v > \tag{21}$$

using the inner-product structure on $C^0(G, \mathcal{F})$.

Looking back on the constant sheaf example, the inner product has that $< c_e, \delta c_v > = c_e^T \delta c_v = (\delta^* c_e)^T c_v = c_e^T \delta^{*T} c_v$. However, all of these quantities are scalars, so in the restricted case of the constant sheaf, one identifies $\delta^* = \delta^T = B$, and one can interpret δ^* as a sheaf generalisation of B. Motivated by equation (12), one can propose a *sheaf Laplacian* $L_{\mathcal{F}}$ by,

$$L_{\mathcal{F}} = \delta^* \delta \tag{22}$$

where δ is the sheaf coboundary operator. This makes sense; δ generalises B which can be thought of as a differential operator, so $L_{\mathcal{F}}$ generalises a second-order differential operator.

One might reasonably ask at this point: what is the point of all of this for PBSHM? The main objective of this paper is to show how a cellular sheaf can provide a rigorous mathematical basis for PBSHM, and this is an application related to the second interpretation of a graph G in Section 3.2, where G is a network graph encoding the population of structures of interest. This idea will be the main subject of the following material. However, it is important to note that cellular sheaves may well prove to be useful for the first type of graphs discussed in Section 3.2; i.e., where G represents an individual structure. Similar technology has already proved of interest in the form of *Graph Neural Networks* (GNNs) [7, 34, 35, 36].

3.5. Sheaf Neural Networks

A GNN is simply a neural network which takes an attributed graph as its input and delivers an attributed graph at its output. The attributes can be defined in relation to nodes, edges and also globally. Typically, the output graph will have the *same topology* as the input graph, but *different attributes*. The application in [7] related to graphs which were the IE models of truss structures; the node and edge attributes were used to encode geometrical and material properties and boundary conditions and one global attribute was assigned as the first natural frequency for normal condition of the truss. A GNN was trained on a large number of simulated truss structures in normal condition so that it could output the first natural frequency. In testing, the GNN could accept an arbitrary truss representation and output the relevant natural frequency. In geometrical terms, considering the first natural frequency as a feature for PBSHM, the GNN could identify the *zero section* of any population made from the same class of trusses. Even more remarkably, it proved possible to add temperature as a global attribute, so that when a truss graph was input with the ambient temperature supplied, the GNN could supply the first natural frequency at a specified reference temperature; the GNN thus provided a *data-normalised zero section*.

One problem associated with GNNs is that they have an inductive bias based on *homophily*; that is they have a built-in assumption that connected nodes in the graph will have similar feature vectors, and this assumption can break down in real-world applications. Furthermore, attempts to build deeper GNNs ran into a problem whereby increasing numbers of layers caused the representations of nodes to converge and become indistinguishable – an issue referred to as *over-smoothing*. In [37], the authors showed that these problems could be alleviated by moving to *Sheaf Neural Networks* (SNNs). SNNs take cellular sheaves as input and produce them as outputs; they provide a generalisation of GNNs in the sense that cellular sheaves generalise attributed graphs.

Bodnar *et al* [37], proposed a learning algorithm for SNNs based on *neural sheaf diffusion*. Consider a sheaf $\mathcal{F}$ defined on a graph G. Now, suppose a set of feature vectors $\underline{x}_v$ defined in the stalks $F(v)$ (note that these define a cochain $\underline{x} = (\underline{x}_1, \ldots, \underline{x}_{N_V}) \in C^0(G, \mathcal{F})$ as defined earlier). One then constructs a matrix X where each column is the cochain corresponding to a feature channel. Now, recall that the main difference between a cellular sheaf and an attributed graph is that the sheaf will have a set of consistent restriction maps; this is the important factor for PBSHM, as SNNs are able to propagate information about transfer maps embodied as restriction maps. Bodnar *et al* proposed a sheaf diffusion process,

$$\dot{X}(t) = -L_{\mathcal{F}} X(t) \tag{23}$$

where $L_{\mathcal{F}}$ is the sheaf Laplacian from equation (22) and $X(0) = X$. This process has the property that it essentially learns the restriction maps for $\mathcal{F}$ from its data over the nodes of G. The diffusion gives the discretised learning algorithm,

$$X_{i+1} = X_i - L_{\mathcal{F}} X_i \tag{24}$$

with $X_0 = X$. Anticipating the next section, Barbero *et al* [38] showed how to use local PCA alignment in order to learn restriction maps by a pre-processing step without iteration.

The current authors anticipate that SNNs will prove effective for SHM problems like inferential measurement, as discussed in Section 3.2; however, this will be a matter for further work, as the rest of this paper will be concerned with cellular sheaves as a conceptual basis for PBSHM.

4. Transfer on Cellular Sheaves

By now, it will be clear that cellular sheaves provide a natural mathematical basis for the PBSHM network of structures and their associated feature spaces. Furthermore, the sheaf structure shows mathematical consistency, unlike the fibre-bundle picture originally used for motivation. However, the sheaf picture discussed so far is of little use analytically, unless some of the main PBSHM operations can be accommodated in the picture. As transfer learning is the main operation in PBSHM, it is important that it fits into the sheaf picture, and ideally should be given a geometric motivation.

The idea of transfer learning in PBSHM can be conveyed effectively by looking at one of the simplest problems. Consider the two-class problem of determining if a given structure S_1 is damaged or undamaged. If labelled training data are available for both classes, one can build a classifier that will then generalise to unseen (testing) data. Further suppose that the classifier is discriminative and works by learning a *decision boundary* in the feature space which separates the data from the two classes from each other. The boundary is learnt from the training data; then as long as any testing data are drawn from the same distribution as the training data, the boundary will serve to identify class labels on the new data. So far, this is standard data-based SHM [1]. Now, suppose that one wishes to classify data from a different structure S_2, but one has insufficient data to train a classifier. If S_1 and S_2 are similar structures and one is using the same features (e.g. first four natural frequencies in each case), one might hope that the classifier for S_1 will work for S_2. This will very seldomly work in practice because of differences in the distributions of the feature data. Anticipating the use of the sheaf picture, the feature spaces for S_1 and S_2 will be denoted $\mathcal{F}(S_1)$ and $\mathcal{F}(S_2)$. Furthermore, the normal and damaged classes for S_i will be denoted by N_i and D_i, and their probability densities by $p(N_i)$ and $p(D_i)$.

For transfer learning, one seeks to use the data from S_1 by more sophisticated means, to improve classification on S_2. Two options immediately present themselves (Figure 9):

1. One somehow constructs a map $\phi : \mathcal{F}(S_2) \longrightarrow \mathcal{F}(S_1)$ with the property that the region of N_2 is mapped onto the region of N_1 and D_2 is mapped onto D_1. In this case, the two normal densities and the two damage densities will coincide in pairs

and the decision boundary established in $\mathcal{F}(S_1)$ will work for both structures. The design of the map ϕ will depend on the availability of data from $\mathcal{F}(S_2)$; there may be no data, sparse data from both classes, sparse or sufficient data from one class; each situation may require a different treatment, but the desired result will be the same. This mode of transfer will be termed *direct transfer*. In the sheaf picture, where the feature spaces are the stalks over the S_i, one example of a transfer map would be the $\tau_{v_1}^{v_2} = (\rho_{v_2}^e)^{-1} \circ \rho_{v_1}^e$ defined earlier, where v_1 and v_2 are the vertices corresponding to the structures S_1 and S_2 and $\partial e = \{v_1, v_2\}$. Note that this map would only work for PBSHM transfer learning if the restriction maps were designed appropriately.

2. One somehow constructs a map ϕ[4], so that $\phi : \mathcal{F}(S_1) \longrightarrow L$ and $\phi : \mathcal{F}(S_2) \longrightarrow L$, where L is a common latent space (often of lower dimension than the $\mathcal{F}(S_i)$) in which (loosely) $\phi(N_1)$ and $\phi(N_2)$ coincide, likewise $\phi(D_1)$ and $\phi(D_2)$. In this case, the classifier is trained *after* the map to L, but only on the $\phi(N_1)$ and $\phi(D_1)$ data; clearly, the classifier will then separate the $\phi(N_2)$ and $\phi(D_2)$. This approach is called *domain adaptation* [6].

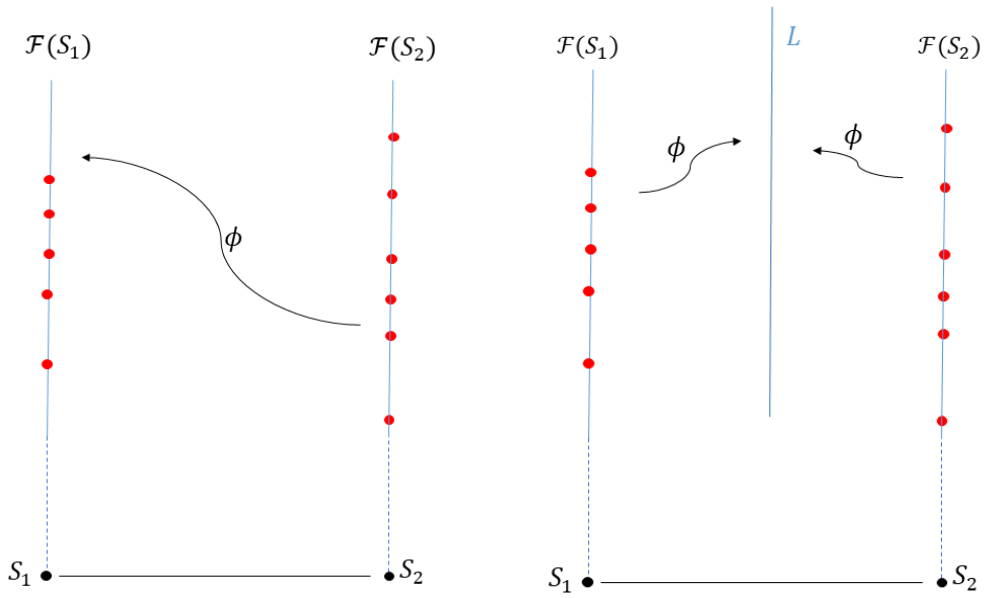


Figure 9: Transfer mechanisms: (a) Direct domain transfer; (b) Domain adaptation.

The rest of this paper will concentrate on the direct transfer approach.

In the completely general problem, there may be a normal/undamaged state N and C damage classes $D^{(1)}, \ldots, D^{(C)}$, leading to a $(C+1)$-class problem. Direct transfer

[4]There is a small abuse of notation here; there are really two maps ϕ_1 and ϕ_2 in the general picture allowed by the sheafs. Following the common notation of domain adaptation, a single ϕ is used here.

would then carry any class data from S_2 into the corresponding class on S_1; a multi-class classifier trained on S_1 would then apply to S_2. Note that the problem now could have many more possibilities for the availability of training data.

In the sheaf picture, the restriction maps ρ_v^e are linear and thus the compositions $\tau_{v_1}^{v_2} = (\rho_{v_2}^e)^{-1} \circ \rho_{v_1}^e$ will be linear. In order to try and pursue a geometrical motivation for transfer here, the analysis will be restricted to linear maps.

Now, it is already the case that a statistical argument has led to a linear transfer map – this is the approach of *Normal-Condition Alignment* (NCA) [32]. This approach assumes that some normal condition data are available from the structures of interest, and some damage-state data are available for S_1. The first step in NCA is to translate the S_2 data so that the mean (centroid) of the normal cluster for S_2 coincides with the mean of the S_1 data; any damage-state data are given the same translation. For the purposes of building classifiers, rigid movements of the feature spaces do not matter, so there is no loss of generality if one removes the normal mean from *all* feature data from *all* structures. This shift automatically carries out the first step of NCA for all structures in the population. Furthermore, in the sheaf picture, the shift places the mean of the normal condition data at the origin of the feature spaces, and thus on the *zero section* of the sheaf, which is then interpreted as the *normal condition section*. So far, this is nice geometry, but the question is: can one implement the later stages of NCA (which can include rotations and scale changes) in a geometrical manner.

Assuming that the feature spaces for S_1 and S_2 have the same dimension, they will be isomorphic, and by an abuse of notation, one can think of the transfer map as,

$$\underline{x}_1 = A\underline{x}_2 \tag{25}$$

where $\underline{x}_2$ is a feature vector in $\mathcal{F}(S_2)$ and $\underline{x}_1$ is its image in $\mathcal{F}(S_1)$ under transfer. The map is linear rather than affine because of the first NCA step – removal of the normal condition mean. A can now be further decomposed as $A = W.O$, where W is a diagonal scaling matrix and O is a rotation. For simplicity, an overall scale w will be adopted here, so that $A = w.O$ where w is a real scalar. If the feature space dimension is D, O will be a member of the orthogonal group $O(D)$. In the picture to follow, w will be stored as a weight on the edge e that joins the vertices v_1 and v_2, representing structures S_1 and S_2; in a sense, this leaves all the transfer maps as rotations. The picture is as in Figure 10 for transfer between two general structures S_i and S_j of the population.

In this view, one can think of the w_{ij} as *affinities* [24] that measure how closely the data resemble each other between $\mathcal{F}(v_1)$ and $\mathcal{F}(v_2)$, after a rotational alignment.

The idea will be adopt aspects of the *Vector Diffusion Map* (VDM) concept from [24].

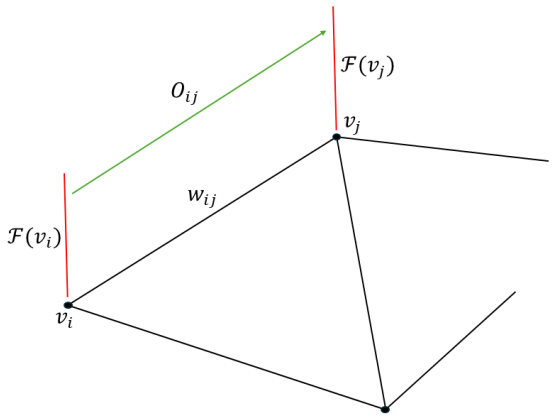


Figure 10: Restricted view of transfer maps as rotations between feature spaces.

5. Vector Diffusion Maps and Transfer

In terms of notation here, the population graph will be denoted $P(V, E)$. The structure S_i will be represented as node $v_i \in V(P)$; the feature data for S_i will be considered to reside in the stalk $\mathcal{F}(v_i)$ of the PBSHM sheaf over P, which is a vector space isomorphic to $\mathbb{R}^D$.

The linear transfer maps here will be based on ideas in the paper [24]; however, the situation in that paper considers data in a Riemannian manifold rather than a vector space, so it will be important to first review the theory there.

5.1. Data on Riemannian Manifolds

The assumption here will be that the data of interest actually lie on (or close to, because of the effect of noise) a Riemannian manifold M of dimension d, where $d < D$ - and hopefully $d \ll D$. The data are assumed to be a set of N points $\{\underline{x}_1, \ldots, \underline{x}_N\}$. The feature vectors will be assumed to be of dimension D, but actually on M, the d-dimensional manifold embedded in $\mathbb{R}^D$. In this case then, the true feature space is a topological space, but not a vector space.

However, locally around a given point $\underline{x}_i$, one can think of nearby points as being in – or nearby to – the tangent space $T_{\underline{x}_i}M$, which can be identified with a d-dimensional hyperplane embedded in the ambient $\mathbb{R}^D$. The first step in the Singer-Wu construction will be to identify the basis for $T_{\underline{x}_i}M$; this will be accomplished by a form of local *Principal Component Analysis* (PCA).

The first stage of the local PCA is to specify a scale parameter ϵ_{PCA}[5]; this is then used to define a neighbourhood $N_{i,\epsilon}$ of $\underline{x}_i$ as an open ball of radius ϵ,

[5]Here, the 'PCA' subscript will be omitted unless confusion is likely to arise.

$$N_{i,\epsilon} = \{\underline{x}_j : 0 \leq ||\underline{x}_j - \underline{x}_i||_{\mathbb{R}^D} < \sqrt{\epsilon}\} \tag{26}$$

The number of points in $N_{i,\epsilon}$ will be denoted by N_i, and the neighbours of $\underline{x}_i$ by $\{\underline{x}_{i_1}, \ldots, \underline{x}_{i_{N_i}}\}$. The neighborhood excludes the point $\underline{x}_i$ itself. Ideally, the neighborhoods should be chosen so that $N_i > d$, but $N_i \ll D$; clearly, this will be dictated by the choice of ϵ[6].

The idea of this construction is that the neighbouring points, will be sufficiently near $\underline{x}_i$, that they are nearby in the embedded hyperplane $T_{\underline{x}_i}M$.

One now defines X_i as the $D \times N_i$ matrix,

$$X_i = (\underline{x}_{i_1} - \underline{x}_i, \ldots, \underline{x}_{i_{N_i}} - \underline{x}_i) \tag{27}$$

which is the feature data matrix for those points in $N_{i,\epsilon}$ shifted by $\underline{x}_i$. For a reasonable number of points in N_i, $\underline{x}_i$ will approximate the mean of the data in the neighbourhood, so the shift will approximate mean removal, which is often the first step in PCA as it simplifies the computation. The idea here is that there will be sufficient data vectors in X_i that they span $T_{\underline{x}_i}M$, so that PCA can then be used to construct a basis.

The next step in the process is to define a function K, which is monotonic and decreasing on the interval $[0, 1]$. Using K, one defines D_i, an $N_i \times N_i$ diagonal matrix specified by entries,

$$D_i(j) = \sqrt{K\left(\frac{||\underline{x}_i - \underline{x}_{i_j}||}{\sqrt{\epsilon}}\right)} \tag{28}$$

which will be used to de-weight points further away from $\underline{x}_i$ and thus most likely to fall out of $T_{\underline{x}_i}M$.

Now, one defines a matrix $B_i = X_i D_i$ on the whole neighbourhood of $\underline{x}_i$; this matrix has a block structure, where the j^{th} block is the data matrix $(X_{i_j} - \underline{x}_i)$ scaled by $D_i(j)$.

The ordered singular vectors of B_i are denoted by $\sigma_{i,1} \geq \sigma_{i,2} \geq \ldots \sigma_{i,N_i}$. Now, if all the points in the neighbourhood $N_{i,\epsilon}$ were to fall exactly in the hyperplane $T_{\underline{x}_i}M$, the first d singular vectors would be non-zero, with the remainder vanishing; in reality, because of noise on the data, it is likely that no singular values will be identically zero. One therefore *estimates* the tangent space dimension d_i by counting the number of singular vectors that account for a percentage variance greater than some threshold γ. One has that d_i is the smallest integer such that,

[6] In fact, Singer and Wu [24] show that a good choice is $\epsilon = O(N^{-2/(d+2)})$, or $\epsilon = O(N^{-2/(d+1)})$ if M is a manifold with boundary.

$$\frac{\sum_{j=1}^{d_i} \sigma_{i,j}^2}{\sum_{j=1}^{N_i} \sigma_{i,j}^2} > \gamma \tag{29}$$

and the first d_i principal vectors are taken as an estimated basis for $T_{\underline{x}_i}M$.

Finally, the estimated dimension $\hat{d}$ of $T_{\underline{x}_i}M$ – and thus M – is taken to be,

$$\hat{d} = \text{median}(d_1, d_2, \ldots) \tag{30}$$

(For simplicity, the notation d will be used later (unless confusion is likely), with the understanding that it is actually $\hat{d}$.)

As is commonly the case, the PCA step will be carried out numerically via a Singular Value Decomposition (SVD) [39]. Suppose the SVD of the matrix B_i is,

$$B_i = U_i \Sigma_i V_i^T \tag{31}$$

and the diagonal Σ_i has the singular values of B_i in decreasing order. Define the matrix,

$$O_i = [U_{i_1}, \ldots, U_{i_{\hat{d}}}] \tag{32}$$

formed from the first $\hat{d}$ left singular vectors (columns) of U_i. By construction, these span $T_{\underline{x}_i}M$, and by the properties of the SVD are orthonormal in pairs; i.e.,

$$O_i^T O_i = \mathbb{I}_{\hat{d}\times\hat{d}} \tag{33}$$

So the construction gives an *orthonormal* basis of $T_{\underline{x}_i}M$. Because of the PCA, the columns of O_i are also the eigenvectors of the covariance matrix associated with the data in $N_{i,\epsilon}$.

5.2. Alignment

The analysis so far has presented a computational method for recovering the tangent space at a point on a manifold from sample data on the manifold; the next stage in the process will be to find a means of aligning nearby tangent spaces; this will lead to a notion of parallel transport and thus to a covariant derivative. As before, the analysis follows that in [24].

Assuming the same data structures as before, suppose that $\underline{x}_i$ and $\underline{x}_j$ are 'nearby' points on the manifold M, but not so close that one is in the constructed neighbourhood of the other. One can adopt a criterion $||\underline{x}_i - \underline{x}_j||_{\mathbb{R}^D} < \sqrt{\epsilon_T}$, where ϵ_T is another scale parameter $\neq \epsilon$. ϵ_T is chosen, as far as possible, so that $\epsilon_T >> \epsilon$, but is small enough that the tangent spaces $T_{\underline{x}_i}M$ and $T_{\underline{x}_j}M$ are close.[7] $\underline{x}_i$ and $\underline{x}_j$ are assumed to be close

[7] Again, Singer and Wu [24] give theory in favour of a value $\epsilon_T = O(N^{-2/(d+4)})$.

enough that the column spaces of O_i and O_j coincide. Now, if the spans of O_i and O_j *were* identical, then there would exist a rotation matrix $O_{ij} \in O(d)$, such that,

$$O_i O_{ij} = O_j \tag{34}$$

so that, on using orthonormality,

$$O_{ij} = O_i^T O_j \tag{35}$$

So that, in this case, O_{ij} is a matrix representation of the operator that transports vectors from $T_{\underline{x}_i}M$ to $T_{\underline{x}_j}M$, viewed as copies of $\mathbb{R}^d$.

In reality, the column spaces will *not* be the same, because of the curvature of M; in this case one redefines O_{ij} as the closest orthogonal matrix in $O(d)$ to $O_i^T O_j$; i.e.,

$$O_{ij} = \overset{argmin}{O \in O(d)} ||O - O_i^T O_j||_{HS} \tag{36}$$

where the norm is the Hilbert-Schmidt norm, given by [40],

$$||A||_{HS}^2 = \mathrm{Tr}(AA^T) \tag{37}$$

The minimisation problem actually has a simple solution, based again on the SVD of $O_i^T O_j$; if $O_i^T O_j = U^{(ij)}\Sigma^{(ij)}{V^{(ij)}}^T$, then the solution to (36) is [41],

$$O_{ij} = U^{(ij)}{V^{(ij)}}^T \tag{38}$$

A schematic of the Singer-Wu construction is shown in Figure 11.

It is shown in [24] (Appendix A) that O_{ij} can be thought of as an approximation to the parallel transport operator from $T_{\underline{x}_i}M$ to $T_{\underline{x}_j}M$, as long as $\underline{x}_i$ is 'close' to $\underline{x}_j$. This operator then defines a covariant derivative on M in the following sense.

Suppose one has a vector field $X(\underline{x})$ defined on M; i.e. a vector $X(\underline{x}) \in T_{\underline{x}}M$ for all $\underline{x} \in M$. Suppose one has a continuous path γ in M, parametrised by t, such that $\gamma(0) = \underline{x}_i$ and $\gamma(t) = \underline{x}_j$, then one can define a covariant derivative ∇, by,

$$\nabla_v X(\underline{x}_i) = \lim_{t \longrightarrow 0} \frac{O_{ij}X(\underline{x}_j) - X(\underline{x}_i)}{t} \tag{39}$$

where v is the tangent vector to γ at $\underline{x}_i = \gamma(0)$.

Now, if this construction is based on data – as above – the immediate problem is that there is no such continuous path, as the operator O_{ij} is constructed on the basis of sampled data; however, if $\underline{x}_i$ is close to $\underline{x}_j$, one can define a discrete approximation,

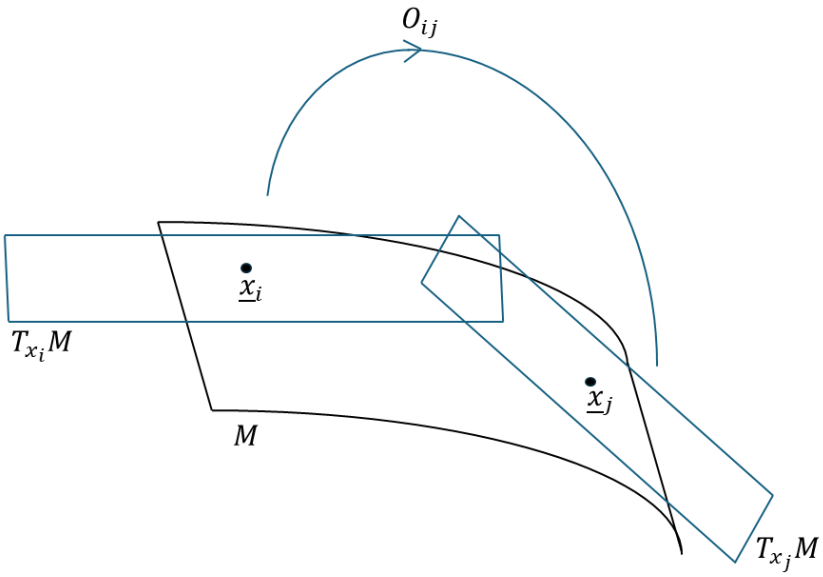


Figure 11: Schematic for Singer-Wu construction of tangent spaces for neighbouring points of data on a manifold and their parallel transport operator.

$$\hat{\nabla}_v X(\underline{x}_i) \approx \frac{O_{ij} X(\underline{x}_j) - X(\underline{x}_i)}{t} \tag{40}$$

where t is the distance along γ from $\underline{x}_i$ to $\underline{x}_j$. Unfortunately, this is not available, as M is not explicitly specified as a manifold. However, at the expense of abusing notation somewhat[8], one could approximate t by $||\underline{x}_i - \underline{x}_j||$, so that,

$$\hat{\nabla}_v X(\underline{x}_i) \approx \frac{O_{ij} X(\underline{x}_j) - X(\underline{x}_i)}{||\underline{x}_i - \underline{x}_j||} \tag{41}$$

which has the property that, if $X(\underline{x}_j)$ is the parallel transport (or aligned value) of $X(\underline{x}_i)$, one has that,

$$\hat{\nabla}_v X(\underline{x}_i) = 0 \tag{42}$$

5.3. Transfer as Alignment on Cellular Sheaves

As before, consider the PBSHM network to be represented by a graph P in which structures S_i and S_j are represented by vertices v_i and v_j respectively. One attaches the data to P via a cellular sheaf $\mathcal{F}$ defined on the graph P. In this case, the feature spaces

[8]The path γ is actually in the manifold, not the tangent space. In the situation where the geometry of M is known explicitly, one would usually define the covariant derivative with γ as the geodesic between $\underline{x}_i$ and $\underline{x}_j$.

for S_i and S_j are identified with the stalks $\mathcal{F}(v_i)$ and $\mathcal{F}(v_j)$. The S_i (resp. S_j) data are thus regarded as points in $\mathcal{F}(v_i)$ (resp. $\mathcal{F}(v_j)$). The transfer problem considered will be one in which S_i is the source structure and has data for all classes – the normal condition data N_i and all relevant damage classes D_i – while S_j will only have normal condition data N_j. The idea will be to align the spaces containing N_i and N_j, and thus implement some geometrically-motivated form of NCA [32]. In fact, the Singer-Wu construction can be almost used directly, potentially with some simplifications.

For the purposes of illustration, the normal condition data N_i and N_j will be assumed Gaussian and their clusters will be represented by ellipsoids in the feature spaces. Because the data are mean-removed, they will be centred on the origin in their respective feature spaces; i.e., on the zero-section of $\mathcal{F}$.

The feature data will usually be expected to be in low to moderate dimension D; e.g. one might monitor the first five natural frequencies of the structure of interest or the magnitudes of 50 spectral lines from an FRF or a transmissibility [1]. In the first case, one might take $m = d = D$ and retain all principal components in the Singer-Wu construction; in the latter case, one might take advantage of the possibility for dimension reduction. The reduction option may well prove important in situations where fewer data are available and it would be difficult otherwise to construct a classifier which could generalise [42].

One can carry out the alignment algorithm just as in the manifold case, but with the N_i (resp. N_j) data treated as the $T_{x_i}M$ (resp. $T_{x_j}M$) data. This will compute an O_{ij} to rotate N_j into N_i. The hope (as always in transfer learning problems of this sort) will be the that a classifier trained on all the S_i data will separate and label the classes of S_j even if testing/new data for S_j represent other classes beyond the normal condition. One can take $\underline{x}_i$ and $\underline{x}_j$ as the nearest points to the origin, or simply the origin, and their neighbourhoods $N_{i,\epsilon}$ and $N_{j,\epsilon}$ as the entire normal-condition data sets; i.e. take ϵ as 'infinite'. If this results in large matrices X_i and X_j, one take small enough subsamples that one does not degrade the estimate of O_{ij}. With these assumptions, one can disregard the need for the weightings D_i and D_j and take $B_i = X_i$ and $B_j = X_j$ (which is also an assumption in [38]. The algorithm then proceeds as before, taking d as a reduced dimension if needed.

The alignment process on normal condition will now be as shown in Figure 12. The discussion will allow a mismatch in overall scale, the mismatch may cause a degradation in transfer (i.e. generalisation of the classifier constructed on S_i data) if substantial, but more of that later.

As defined above, this is a method of *direct transfer*, where one has gone directly to the map $\tau_{v_j}^{v_i}$ without constructing or defining the sheaf restriction maps. The approach is summarised in the schematic shown in Figure 13. Local PCA alignment on cellular sheaves is discussed in [38] in the context of sheaf neural networks generally, rather than for transfer learning.

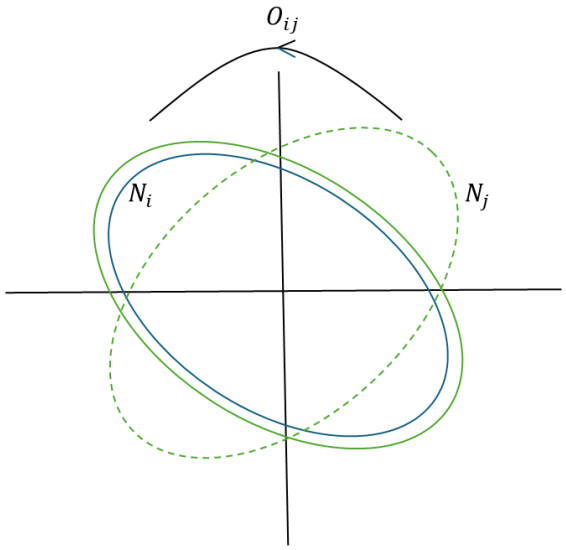


Figure 12: Effect of rotation O_{ij} on N_j normal-condition cluster (green), showing alignment with N_i cluster (blue); initial position of N_j is shown as a dotted line.

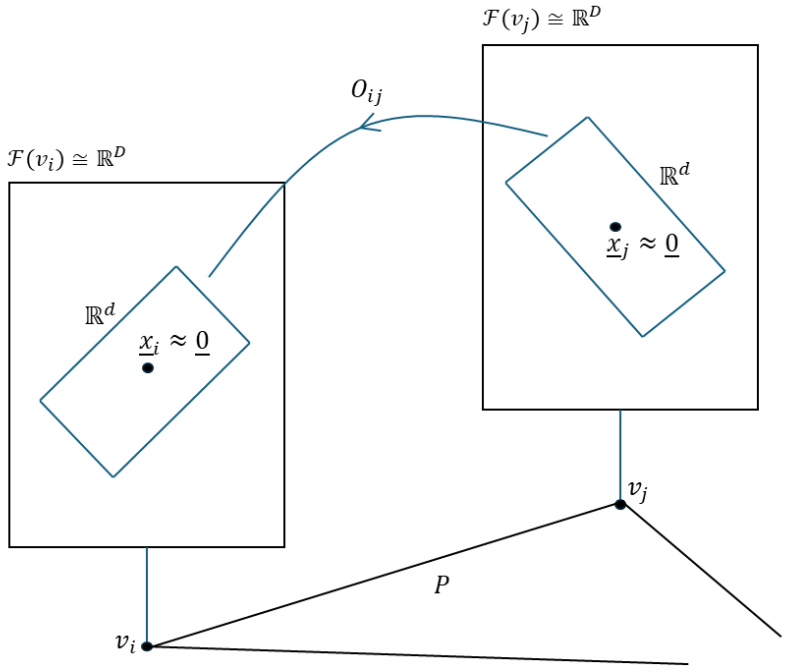


Figure 13: Schematic for transfer realised as alignment on a cellular sheaf.

The approach here realises the aim of generating transfer naturally from the geometry of PBSHM. In fact, the alignment mapping from $\mathcal{F}(v_j)$ to $\mathcal{F}(v_i)$ can be thought of as a direct analogy of parallel transport of a vector field across a manifold, and as arising from a discrete covariant derivative,

$$\hat{\nabla}_v X(\underline{x}_i) \approx O_{ij} X(\underline{x}_j) - X(\underline{x}_i) \tag{43}$$

The transfer algorithms obtained from the geometry are discussed in the next section along with the scale mismatch noted earlier and illustrated in Figure 12.

5.4. Affinities and Statistic Alignment

A number of linear transfer maps are discussed in [32], which exploit the idea of harmonising lower-order statistical moments between source and target data sets – the idea of *statistic alignment* (SA) [43]. [32] investigates the use of SA for transfer learning in PBSHM and examines a range of methods. Before discussing these approaches, it is important to reiterate what is meant by the usual operation of *standardisation*; this is an operation on data which attempts to give equal emphasis to each feature in a feature vector. Suppose one has training data for the source problem $\{\underline{x}_i^s; i = 1, \dots, N_i^s\}$ and for the target problem $\{\underline{x}_i^t; i = 1, \dots, N_i^t\}$[9]. In general, these data might represent a number of classes as discussed in Section 4; however, here they are *only* taken to represent the normal condition data as damage-state data may be completely absent in the training data for the target problem. Now, denote the j^{th} component of the i^{th} data point by x_{ij}^s (resp. x_{ij}^t) of the source (resp. target problem). Furthermore, denote the columnwise means and standard deviations of the source (resp. target) dataset by μ_j^s and σ_j^s (resp. μ_j^t and σ_j^t). Standardisation of the source and targets separately is accomplished via the transformations,

$$z_{ij}^s = \frac{x_{ij}^s - \mu_j^s}{\sigma_j^s} \tag{44}$$

$$z_{ij}^t = \frac{x_{ij}^t - \mu_j^t}{\sigma_j^t} \tag{45}$$

Following these transformations, the source and target distributions are harmonised in the sense that each z-component has zero-mean and unit standard deviation; however, this only ensures perfect agreement if the covariance matrices for source and target data are diagonal. Following Poole *et al.* [32] this approach will be referred to here as $A - standardisation$. This transformation may be benchmarked against a standard data-based SHM approach where one has source and target data, but simply aggregates

[9] In order to train a classifier, one of course also needs label data for at least the source problem; however, these labels are not used in constructing the transfer maps.

them to build a classifier; this will be referred to as $N-standardisation$. Reference [32] also considers more sophisticated approaches:

- **Normal-Condition Alignment (NCA)**: In this approach the target data are first mean-shifted so that they are centred on the source mean and then scaled to agree in standard deviation. The source data are unmoved, and the target data are subject to,

$$z_{ij}^t = \frac{x_{ij}^t - \mu_j^t}{\sigma_j^t}\sigma_j^s + \mu_j^s \tag{46}$$

  which is clearly simplified if all the population data have been already shifted to have zero-mean. This approach is related to A-standardisation but retains the statistics of the source problem, so should be thought of as a means of *direct transfer*. In contrast, A-standardisation harmonises the statistics of source and target data in a different domain to both and can be thought of as means of *domain adaptation*.

- **Correlation Alignment (CORAL)**: This approach, originating in [44], attempts to deal with correlations in the feature data by first applying $A-standardisation$ and then computing a rotation of the source domain to harmonise the covariance matrix of the data with that in the target domain.

- **NCORAL**: Follows NCA by a rotation to harmonise the covariance matrices in source and target; the approach thus considers a mean-shift a scale change and a rotation.

All of these methods were compared on a PBSHM problem in [32]; a population of two numerically-simulated three-storey structures was considered, with very large deviations in their dimensions and material properties. The damage states corresponded to reduction of stiffness in three locations; together with the normal condition, this led to a four-class SHM problem. The features used were the first three damped natural frequencies and the classifier used was a simple k-nearest neighbour approach. The results were excellent; only N-standardisation gave poor results, the other SA methods gave perfect classifiers. Somewhat surprisingly, a number of nonlinear domain adaptation methods failed to improve on N-standardisation; this appeared to be a result of the large differences in mean and covariance between source and target thwarting the nonlinear optimisations concerned. This indicated that linear SA methods should be applied as an initial step in more challenging transfer problems to give the nonlinear methods good initial conditions for their optimisations. Another benefit of the linear methods is that they only estimate first and second-order statistics and so do not have very high demands on the size of the training set. Furthermore, the paper showed that NCORAL could give excellent results on a challenging real-life problem in transferring between SHM problems for two bridges – the Z24 bridge [45] and KW51 bridge [45]. The data for the two bridges even encompassed environmental variations.

Having shown that linear transfer methods actually have considerable value in PBSHM

problems; it is worth proceeding to discuss the geometrical motivations for the methods. Before explaining how the Singer-Wu constructions allow this, it is necessary to consider a further aspect of their analysis – that of *affinities*. The idea relates to the scale mismatch already discussed and shown in Figure 12.

The discussions needs to return to the original Singer-Wu construction for data distributed in a Riemannian manifold M as covered in Section 5.1; one observes that the construction actually induces a natural (part of a) cellular sheaf. One might consider the set of data points to define a graph, where each point represents a vertex. For large data sets, this would be a complex graph with many nodes; however, not all of the potential connectivity is important. Not all points are used to generate a tangent space; in fact, only those points $\underline{x}_i$ identified to have a neighbourhood $N_{i,\epsilon}$ are associated with an approximate $T_{\underline{x}_i}M$, and roughly speaking, those points will be about $\sqrt{\epsilon_T}$ apart. This subset of the data will be said to define the vertices of a graph G_{ϵ_T}. If two points $\underline{x}_i$ and $\underline{x}_j$ are nearby enough (and also far enough away from each other) that the bases for the tangent spaces $T_{\underline{x}_i}M$ and $T_{\underline{x}_j}M$ are defined, and are aligned – i.e. so that a parallel transport operator O_{ij} exists – then the edges e_{ij} corresponding to pairs v_i ($\underline{x}_i$) and v_j ($\underline{x}_j$) are said to exist. If two vertices in G_{ϵ_T} have bases that are not aligned, there is no edge between them. By construction, each of these vertices v_i has a vector space attached to it – the tangent space $T_{\underline{x}_i}M$ – which can be interpreted as the stalk $\mathcal{F}(v_i)$ for some cellular sheaf over G_{ϵ_T}. The restriction maps are not uniquely defined, but they are specified up to composition as one can make the identification $\tau_{v_j}^{v_i} = O_{ij}$. The stalks $\mathcal{F}(e_{ij})$ are not uniquely fixed by this construction either.

Now, recall that because of the curvature of M (and potentially because of noise on the data), the constructed maps O_{ij} will not perfectly align the tangent space bases, with the error increasing with the distance between the points $\underline{x}_i$ and $\underline{x}_j$. One can therefore associate a 'mismatch factor' w_{ij} with each pair of vertices (v_i, v_j) in G_{ϵ_T}, based on their distance apart in M or in the embedding space $\mathbb{R}^D$. To make things quantitative, one could introduce a monotonically-decreasing kernel function K (which need not be the same as the one used in the local PCA), and then define,

$$w_{ij} = K\left(\frac{||\underline{x}_i - \underline{x}_j||_{\mathbb{R}^D}}{\sqrt{\epsilon_T}}\right) \tag{47}$$

However, this is still not quite correct; one needs K to be only supported on the interval $[0, 1]$ and zero elsewhere, because the construction demands that bases only be aligned if $0 < ||\underline{x}_i - \underline{x}_j|| < \sqrt{\epsilon_T}$. So, if one desired to use a Gaussian kernel, one would actually need to specify $K(u) = e^{-u^2}\chi_{[0,1]}(u)$ where χ is the characteristic function for $[0, 1]$ i.e. is unity if $u \in [0, 1]$ and zero otherwise. With this choice, one clearly has the $w_{ij} = 0$ whenever $||\underline{x}_i - \underline{x}_j|| > \sqrt{\epsilon_T}$.

The weights w_{ij} can now be formally attached to the edges $e_{ij} \in E(G_{\epsilon_T})$ making it a weighted graph.

Singer and Wu then define a matrix S_{ij} by,

$$S_{ij} = \begin{cases} w_{ij} O_{ij} & e_{ij} \in E(G_{\epsilon_T}) \\ 0 & \text{otherwise} \end{cases} \tag{48}$$

which is a key element of the construction of objects called *vector diffusion maps*. One needs to go no further with the analysis to draw the analogy with transfer learning; for those interested, the paper uses the VDMs as key components of proofs that show that the graph Laplacian on G_{ϵ_T} converges to the continuous-manifold Laplace-Beltrami operator as the number of sample points $\underline{x}_i$ tends to infinity (very roughly speaking). This analysis reinforces the interpretation of the operator $\hat{\nabla}$ derived earlier as a finite-difference approximation to a true covariant derivative.

One can now draw the analogy aimed for. Consider the cellular sheaf defined as the PBSHM framework. It is shown above that one can use a Singer-Wu-like construction to construct transfer maps O_{ij}, which are essentially parallel transport operators associated with a discrete covariant derivative. However, because the O_{ij} are rigid rotations, they cannot compensate for scale differences in the source and target feature spaces – they are essentially the analogue of the CORAL algorithm discussed above, which is purely motivated by statistics. To improve matters, the transfer map needs scale parameters. Suppose that a single overall scale w_{ij} served to compensate, then the transfer map would become $w_{ij}O_{ij}$ rather like in the definition of the S_{ij}. The NCORAL algorithm discussed above actually allows more general maps. Now, to make the analogy between transfer maps and the S-matrices even more interesting, consider the discussion in Section 3.2 about the connectivity of the population graph P. In that discussion two vertices v_i and v_j were said to be connected if positive transfer were probable for the data of the two structures. In [31], the authors show that for a specific PBSHM population, it is possible to compute the *probability of positive transfer* p_{ij} between the feature spaces of two structures S_i and S_j; the idea being that one would only allow transfer if $p_{ij} \geq 1/2$; these probabilities could then be used to weight the edges of the population graph, removing any edges for which $p_{ij} < 1/2$. (The value $1/2$ for the threshold here assumes that positive and negative errors have equal costs; this can be adjusted if the costs are different, as is likely to be the case for an SHM problem.) This not only fixes the topology of the population graph P, but it also allows a general definition of the transfer maps τ_{ij} by,

$$\tau_{ij} = \begin{cases} A_{ij} O_{ij} & p_{ij} \geq 0 \\ 0 & \text{otherwise} \end{cases} \tag{49}$$

where the A matrix now compensates for any error from the rotation caused by scale mismatch; such a matrix being specified by the NCORAL algorithm.

One can now see how linear transfer maps arise from the geometry of cellular sheafs via a Singer-Wu-like construction; in increasing order of sophistication:

1. The simplest option is to harmonise the means of the source and target feature data. Without loss of generality, one can take (by translation) the mean of the

normal-condition data of a reference feature set to be zero. The transfer maps then place all the normal-condition data on the zero-section of the cellular sheaf representing the PBSHM population.

2. One allows rotations in order to harmonise source and target data; this is as in the CORAL algorithm. The transfer maps then correspond to parallel-transport operators corresponding to a discrete covariant derivative.
3. One allows translation, rotation and scaling – the most general linear map – as in the NCORAL algorithm. This corresponds to an analogy of the S_{ij}-matrix construction in Singer and Wu's definition of a vector diffusion map.

The geometrical content of the first two of these options is now clear; the third is much more of an analogy and could be the subject of future work. While the Singer-Wu procedure provides a map between feature spaces which encodes many of the properties previously identified as important for linear transfer, its construction is purely geometrical and it is not guaranteed to be *optimal* for transfer. Furthermore, it is different in some respects to previously-proposed approaches; for example, it differs from NCORAL in the order of the scaling and rotation operations, and as these operations do not commute, the algorithms will be expected to differ. Nonetheless, as the new construction shares many of the features of existing statistic-alignment methods, it is expected that it will effect a reasonable degree of transfer. The efficacy of the new algorithm is investigated in the next section via an SHM case study based on numerical simulation.

6. Case Studies

To examine the performance of the linear transfer approach derived in Section 5.3, the algorithm was applied to a numerical case study previously presented in [31, 46]. The case study considered centres on a population comprising 20 simulated ten-degree-of-freedom (ten-DoF) mass-spring-damper systems, examples of which are shown in Figure 14. Values for the material and geometrical properties for the numerical case study are given in Table 1. This paper considers homogeneous TL, where the label spaces of the source and target problems are equivalent, so only damage in the base structure – the linear chain – was considered. To generate a meaningful population, random variants of the base were generated by adding one to three connections to ground from the masses; these connections were made randomly to the six central masses in the chain. Each lumped mass was assumed to be a rectangular volume, parameterised by a length l_m, width w_m, thickness t_m, and density ρ, with the density sampled from a Gaussian distribution to represent manufacturing variation. The stiffness between masses was assumed to follow the model for the tip stiffness of a cantilever beam $k_b = 3EI/l_b^3$; the dimensions of the beam being denoted $\{l_b, w_b, t_b\}$. The elastic modulus E was also drawn from a Gaussian distribution at first, but initial trials showed that representative variability could be introduced via the density alone. The stiffness for the ground connections was modelled following the same approach. Damping c was not derived from a physical model, instead, it was drawn from a Gamma distribution directly. Damage at a given degree of freedom

was modelled as an open crack, located in the centre of each connection, with a length of 0.1 m. It was modelled as a reduction in stiffness following the model presented by [47]. The nominal values for the parameters and their variations are given in Table 1. For further explication of the ten-DoF case study, the reader is directed to [46].

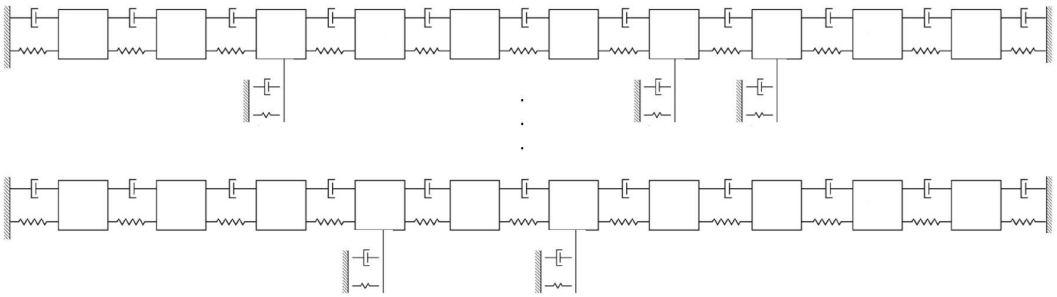


Figure 14: A sample of structures from a population of ten degree-of-freedom systems with up to three random connections [46].

	Unit	Values
Beam geometry $\{l_b, w_b, t_b\}$	m	$\{5.6, 1.1, 6\}$
Mass geometry $\{l_m, w_m, t_m\}$	m	$\{5.6, 1.1, 6\}$
Crack geometry $\{l_{cr}, l_{loc}\}$	m	$\{0.1, 2.8\}$
Elastic modulus E	GPa	20
Density ρ	kg/m^3	$\sim \mathcal{N}(2300, 20)$
Damping coefficient c	Ns/m	$\sim \mathcal{G}(8.0, 0.8)$

Table 1: Properties for the numerical case study.

The population can be regarded as *weakly heterogeneous* because, although each base individual belongs to the same class of ten-DoF lumped-mass systems, the additional connections to ground alter the structural topology [5] and induce a substantive change in the dynamics.

The healthy, or undamaged condition, for each system was represented by taking all stiffnesses to have their original value. Damage conditions were simulated by reducing the stiffness values for the middle nine springs by 50%, one at a time, resulting in nine distinct damage classes. To ensure consistency in the label spaces between individuals, the connections to ground were not subjected to damage. Natural frequencies and modal matrices were obtained from parameter matrices using the standard eigendecomposition. Modeshapes were then normalised to unit length.

For each system in the population, labelled datasets were generated by sampling natural frequencies via the process described above. These features were provided with the corresponding health-state label, denoting either the undamaged condition, or one of the nine damage locations. Two modes of classification were considered: binary and multiclass. For binary classification, 500 datapoints were generated for the undamaged

class and 50 points were generated for each of the damage classes, yielding 950 in total. For multiclass classification, 500 datapoints were generated for all classes, yielding 5000 points in total. This data generation process ensured reasonably balanced class representation within the two datasets. Scatter plots showing the first two natural frequencies and the first two principal components for a representative dataset are provided in Figures 15 and 16, respectively. Note that the PCA projection appears to show greater class overlap; this is entirely to be expected and does not represent the separability of classes in the full feature space.

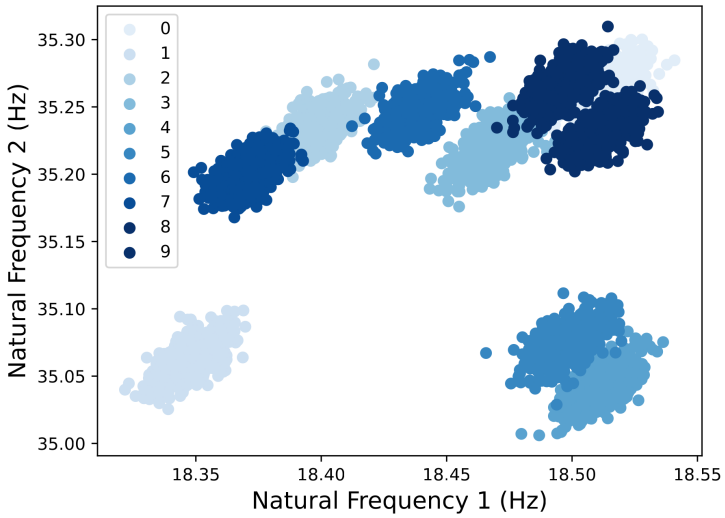


Figure 15: Distribution of the first two natural frequencies for a ten-DoF system under varying damage conditions. State 0 corresponds to the undamaged condition and States 1–9 correspond to damage conditions.

6.1. Classification and Transfer Tasks

From the datasets generated, 380 unique transfer-learning scenarios were created by taking each system in turn to be a source domain and treating the remaining 19 systems as target domains. For the purposes of transfer, it was assumed that label information were available for all datapoints associated with the source domain, but only labels corresponding to the healthy state data were available for the target domain.

For each target domain, classification tasks were defined in which labels $y_i \in Y$ were to be predicted from the natural frequencies of the system $\mathbf{x}_i \in X$, where $X = \mathbb{R}_+^{10}$ and where $Y = \{0, 1\}$ for the binary case, and $Y = \{0, 1, 2, ..., 10\}$ for the multiclass case. Here, $y = 0$ denotes a healthy state and $y > 0$ denotes a damage state. To serve as a classifier mapping $f : X \to Y$, a k-nearest-neighbour (kNN) classifier was used with $k = 3$. At this stage, it is important to note that the target domain features, i.e. the natural frequencies, were permuted to maximise the trace of the modal assurance

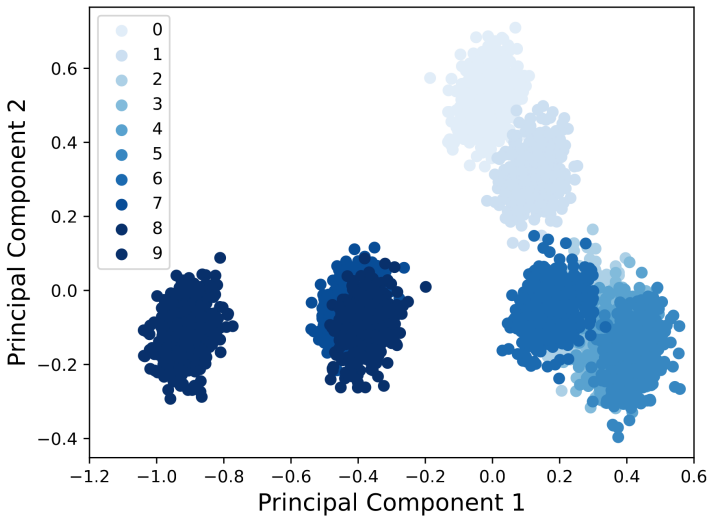


Figure 16: Distribution of the first two principal components for a ten-DoF system under varying damage conditions. State 0 corresponds to the undamaged condition and States 1–9 correspond to damage conditions.

criterion (MAC) between the modeshapes of the source domain and the target domain in their respective undamaged states. This correction was considered necessary to avoid damage causing modes to swap.

To assess the transfer outcomes, classifiers were first trained on the source-domain data. Subsequently, the transfer mapping was learned using the undamaged-state data for both the source and target domains, and then applied to the target domain data. The transformed data for the target domain were then provided as inputs to the classifier and the predictions were evaluated using the accuracy, which was calculated as follows,

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{50}$$

where TP denotes the number of true-positive predictions, TN denotes the number of true-negative predictions, FP denotes the number of false-positive predictions (type-I errors) and FN denotes the number of false-negatives (type-II errors).

To benchmark the new algorithm and provide a basis for comparisons, NCA was used as a reference transfer algorithm [46]. Additionally, to allow one to examine the trends in accuracy between domains differing in similarity, a data-based proxy for structural similarity was used. This similarity measure was defined as the normalised trace of the permuted MAC matrix for undamaged conditions of the source and target domains, in accordance with the previous presentations of the case study [31, 46]. The similarity scores between structures are presented in Figure 17, which provides a visualisation of

the heterogeneity within the population.

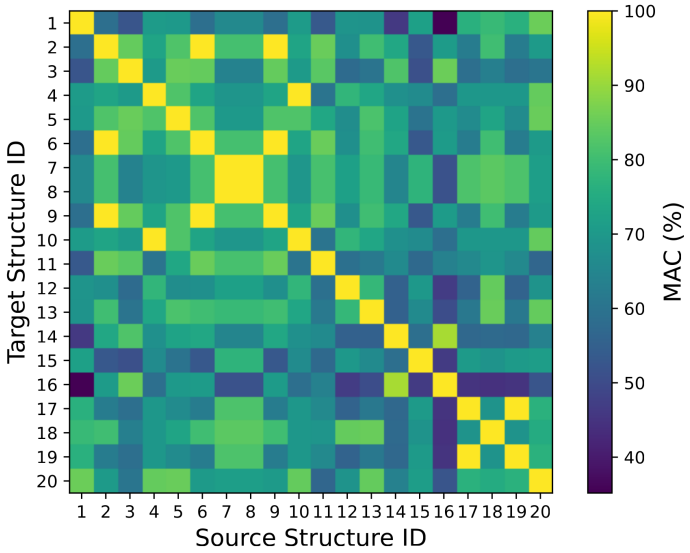


Figure 17: Structural similarity scores between the 10-DoF systems, derived from the MAC.

6.2. Results: Binary Classification

The first case study simply looked at separating normal-condition data from damage-state data – a binary classification problem. Figure 18 shows classification accuracies as a function of similarity (MAC) for each of the 380 transfer tasks, with the proposed PCA-alignment variant shown in orange, and NCA shown in blue. Both transfer approaches perform very strongly, exceeding 97% accuracy. Nonetheless, it can be seen from Figure 18 that, on average, NCA performs slightly better with a mean accuracy (shown in Figure 18 as solid lines) of 99.3%, compared to a mean accuracy of 97.4% for the proposed PCA variant. A possible explanation for this discrepancy is provided in Section 6.4. The important point to note is that the geometrically-motivated algorithm produces very good results.

It is also apparent From Figure 18, that the binary-classification accuracies post-transfer show very little correlation with the MAC-based proxy for structural similarity. This is to be expected, as both transfer approaches in question align the normal conditions between the source and target, and variations in the classification performance are largely dominated by the degree of overlap between the clusters for the normal condition and any of the damage conditions in the transformed target domain – this factor is independent of the source domain and is therefore also independent of the similarity between the source and target domain.

Tables 2 and 3 show the normalised confusion matrices for the binary classification averaged over the 380 transfer tasks for the PCA variant and NCA respectively. Again,

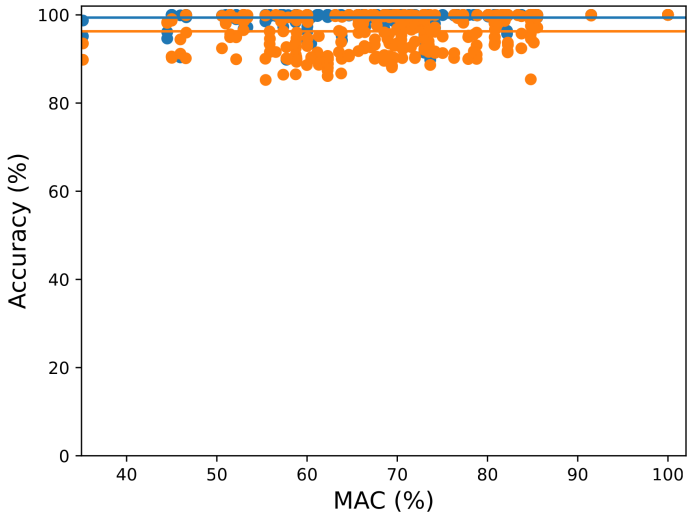


Figure 18: Binary classification accuracies post-transfer the PCA variant (orange) and NCA (blue) against the average MAC value between source and target domain. Solid lines denote the mean accuracy.

Target Class	Predicted Class 0	Predicted Class 1
0	0.53	0.0
1	0.026	0.45

Table 2: Mean confusion matrix for binary classification following transfer achieved using the proposed PCA variant.

Target Class	Predicted Class 0	Predicted Class 1
0	0.53	0.0
1	0.0066	0.47

Table 3: Mean confusion matrix for binary classification following transfer achieved using NCA.

it can be seen from these matrices that both transfer algorithms have strong performance in terms of true predictions. The misclassifications arising from both transfer algorithms are type-II (false-negative) errors, meaning data corresponding to damaged classes were misidentified as healthy. This result indicates that, in the alignment processes, damage data are brought into proximity with the undamaged class; because of the higher relative density of the undamaged class (the damage data are dispersed across multiple clusters), these overlapping damage data are more likely to be misclassified.

6.3. Results: Multiclass Classification

The proposed transfer approach was next tested for a much more challenging ten-class classification task. Figure 19 shows multiclass classification accuracies for each of the 380 transfer tasks, again with the proposed PCA variant in orange, and NCA in blue. It is apparent again (Figure 19) that NCA yields a stronger performance than the proposed PCA variant, with a NCA achieving a mean accuracy of 55.3%, and the PCA variant achieving a mean accuracy of 36.7%. In both cases, the transfer algorithms yield a benefit if compared to a random classifier, which would in expectation achieve an accuracy of 10%. Moreover, both transfer approaches yield marked improvements beyond a classifier that correctly identifies all data in the aligned normal conditions but classifies randomly for the damage states – a classifier with this behaviour would achieve an expected 21% accuracy. The figure shows that the PCA-alignment algorithm sometimes reduces to a binary classifier (10% accuracy), but also produces the best classifier of all, at the highest level of MAC similarity.

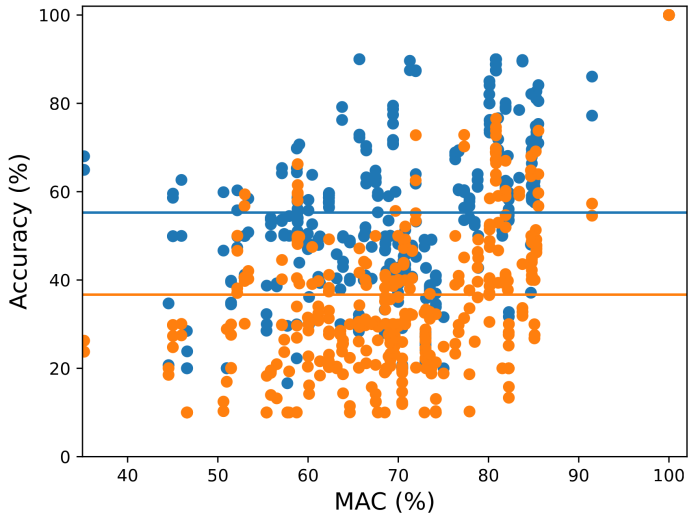


Figure 19: Multiclass classification accuracies post-transfer the PCA variant (orange) and NCA (blue) against the average MAC value between source and target domain. Solid lines denote the mean accuracy.

Unlike in the binary classification case, the multiclass setting shows some correlation between the MAC-based proxy for structural similarity and the accuracy discernable

from Figure 19. Specifically, there appears to be a weakly-positive correlation between the two variables. This result is to be expected as systems that are more similar in terms of their structure, and therefore their dynamics, are likely to be affected by localised reductions in stiffness in similar ways. This effect results in clusters representing damage conditions appearing in similar locations relative to the healthy condition, for similar source and target domains.

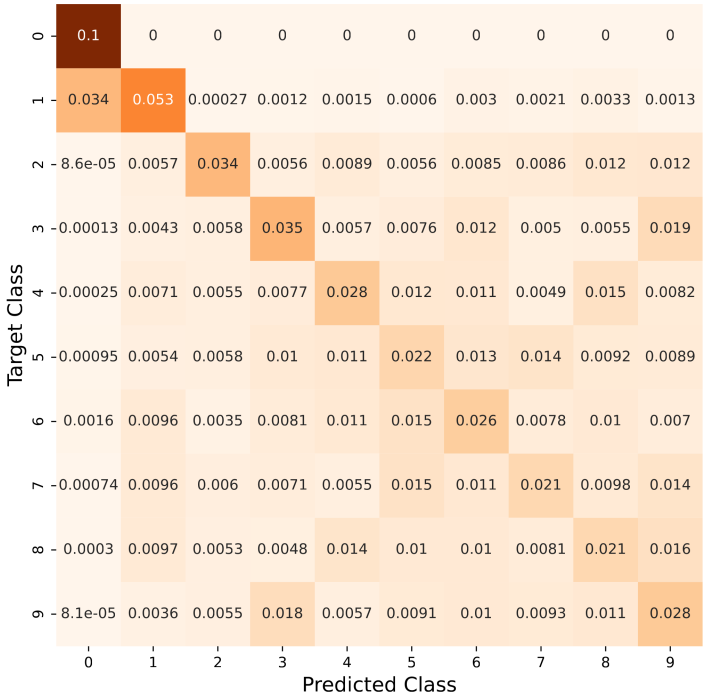


Figure 20: Mean confusion matrix for multiclass classification following transfer achieved using the proposed PCA variant.

Figures 20 and 21 show the normalised confusion matrices for multiclass classification averaged over the 380 transfer tasks for the PCA variant and NCA, respectively. As with the binary case, both transfer approaches show no instances of the undamaged class being misidentified as damaged. Rather, all misclassifications are a result of mislabelling a damage case as healthy, or as confusing two damage labels. In the multiclass setting, both transfer approaches retain dominant diagonals in the confusion matrices, further indicating that transfer aids in the classification of localised damage. Some amount of confusion between different damage states can possibly be attributed to symmetries in the ten-DoF systems; for example, in a completely symmetrical system, a reduction in stiffness at Spring 2 will manifest in the natural frequencies in the same manner as

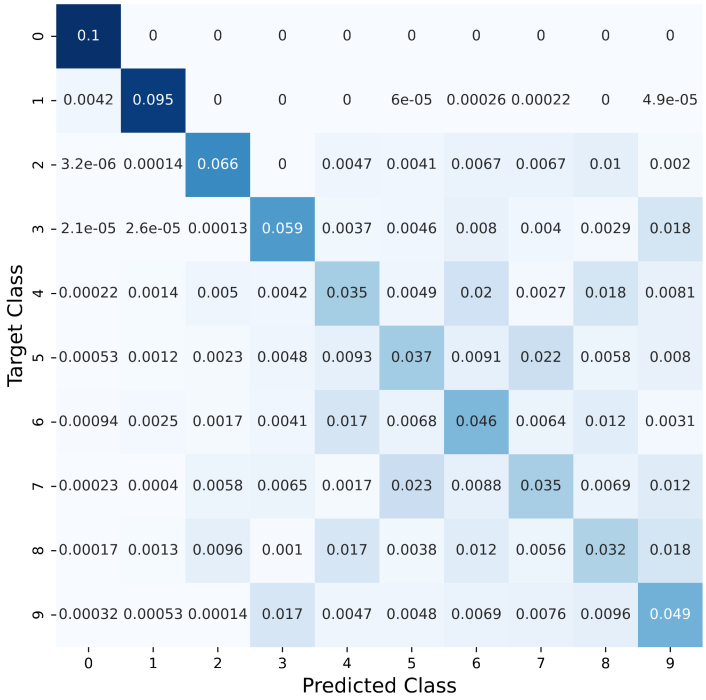


Figure 21: Mean confusion matrix for multiclass classification following transfer achieved using NCA.

an equivalent reduction in stiffness at Spring 10. The symmetry here is broken by the variations in stiffness etc. and by the additional connections to ground.

Similar results where more simple statistic alignment approaches outperform more involved domain adaptation approaches have been reported in previous works. For example, in [32], NCA yields more favourable transfer results compared to transfer component analysis and domain adaptation neural networks in a case study centred on three-DoF mass-spring-damper systems.

6.4. Interpretation

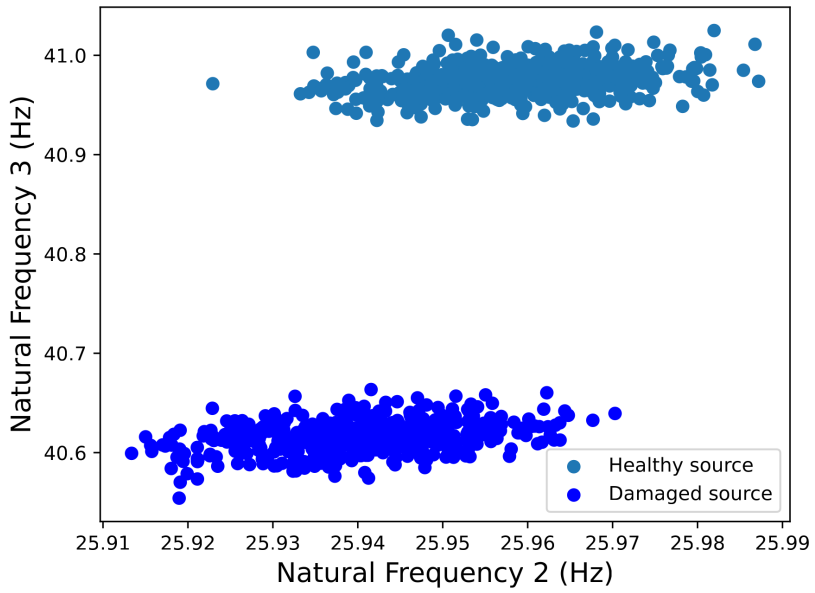
From the results of the binary and multiclass classifications presented in the previous subsections, it is evident that, for the current numerical case studies, the proposed variant of PCA alignment produces good results, which are nonetheless a little inferior to those from NCA. Given that the sole difference between the PCA variant and NCA is the rotation step present in the PCA variant, these results represent a valuable opportunity to understand the effect of rotations in linear transfer learning for population-based SHM, and particularly for transfer in the absence of damage-state data.

The closer look at the effect of rotations here will consider an illustrative two-class sub-problem from the numerical ten-DoF system case study, in which System 4 is considered as the source domain and System 20 is considered as the target domain. These structures possess high similarity (greater than 80%) according to the MAC metric; however, they yielded a relatively-poor post-transfer accuracy in the classification task (approximately 80% for the binary case). To allow a visual understanding of the transfer process, a low-dimensional subset of features was selected; specifically, the second and third natural frequencies. Additionally, a single damage state was considered for clarity.

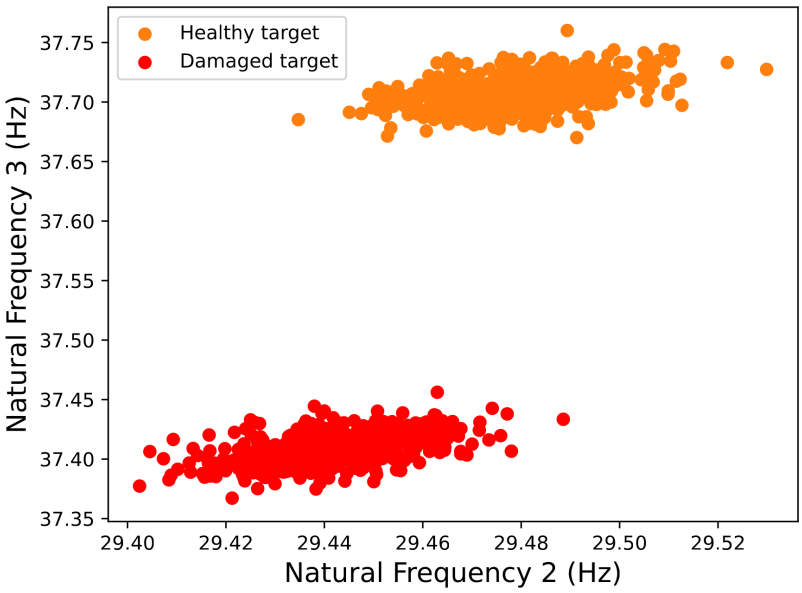
The data distributions in the original feature spaces for the source and target domain are presented in Figures 22(a) and 22(b), respectively. It is immediately apparent that the joint distributions over features and labels $P(X, Y)$ are similar between source and target domain; thus, it would be reasonable to expect high transfer performance between these domains. In fact, because all of the principal axes of the various classes appear to be nearly parallel and the relative positions of the normal and damage classes are similar, the transfer can be accomplished by translation alone and so is particularly suited to NCA.

Figure 23(a) shows the transformed domains following the alignment of the normal conditions using NCA. For this transfer approach, the excellent classification accuracy that one would expect is, in fact, achieved. The almost perfect alignment between the healthy clusters induces very good alignment between the damage clusters.

Figure 23(b) shows the transformed domains following the alignment of the normal conditions using the PCA variant. Here, it can be seen that, again, the normal conditions have excellent alignment. However, the process of aligning the principal directions of the normal conditions has rotated the target damage cluster away from the source damage cluster so that is now *closer* to the source healthy condition. This effect, of course, results in misclassifications when using a kNN classifier. While the problem would not occur with a discriminative classifier, it is important to acknowledge it. At root, the problem can be explained again geometrically and may prove to be an issue with rotations. One can think of there being two main sets of directions in the source domain; firstly there are the principal directions (components) of the normal-condition data, and secondly there are the directions of the damage classes relative to the healthy cluster. These directions need not be the same. Now, one can imagine a situation where the normal-damage direction is the same in the source and target domains, but the normal cluster in the target domain is rotated locally with respect to that in the source domain. In this case, a rotation in the alignment process will produce exactly the effect shown in 23(a). In fact, in the simplified problem considered here, there appeared to be good alignment between all the clusters in Figures 22(a) and 22(b); however, this effect was an artefact of the axis scalings in those plots. When the various clusters are considered 'at a distance' as in Figure 23(b), it is clear that the clusters are nearly isotropic, *not* markedly elliptical, and a significant rotation is used in the PCA-based algorithm to bring the normal clusters into alignment. In a sense, the worst case analysis will potentially occur when both the source and target healthy distributions are isotropic, the rotation step will be very

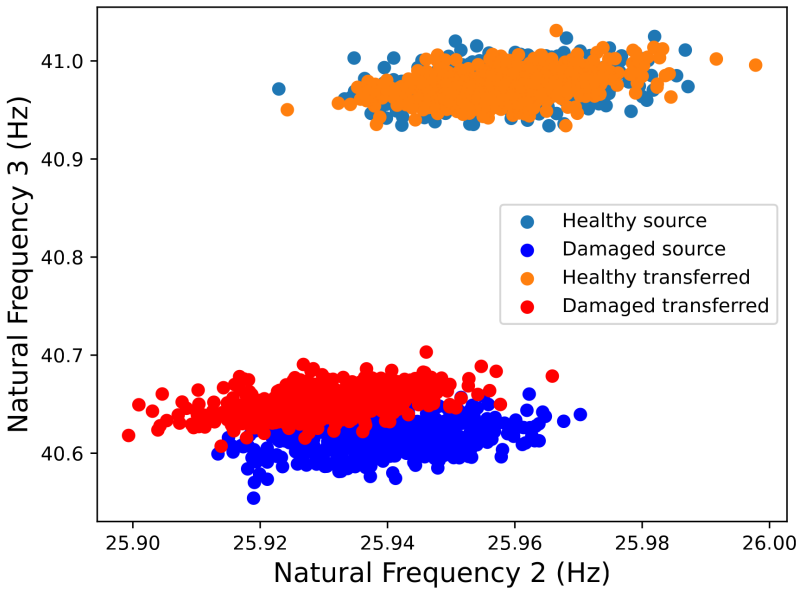


(a)

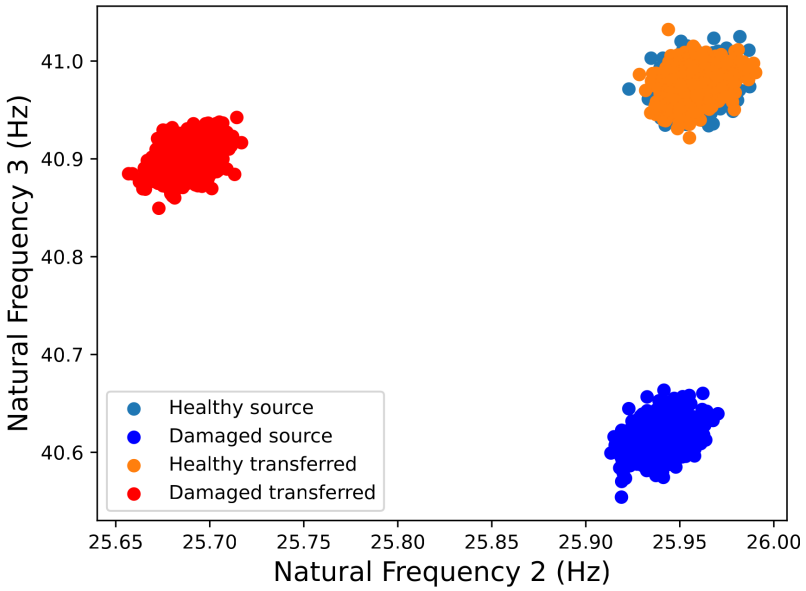


(b)

Figure 22: Natural frequency distributions for the undamaged condition and damage condition for: (a) System 4 (source domain), (b) System 20 (target domain).



(a)



(b)

Figure 23: Natural frequency distributions post-transfer using: (a) NCA, (b) PCA variant.

sensitive to any samples far from the mean; the process is thus not robust.

The problem of conflicting directions in SHM feature spaces is already well-known in the context of operational and environmental variations [48]. There, the problem is that a variation with temperature, for example, would result in an elongation of the normal-condition data in a certain direction, and this can reduce sensitivity to damage. One solution to the problem is to project out the 'temperature direction'; however, if the damage-states are in the same direction, they will be removed too. Of course, if one has no damage-state data – and this is the major motivation for transfer learning for SHM – there is little one can do. One possibility is to choose the classifier carefully; as mentioned earlier a discriminative margin classifier, in which classification decisions are made based on distances from a learned boundary rather than distances from other data points or data centroids, could provide improved robustness to the pathological displacements introduced by rotations.

There are several conditions that may exacerbate the impact of rotations; it is worth being cognisant of these conditions when using affine transformations for domain adaptation.

The first consideration that could be made, regards the centre of rotation for an affine transformation. In many SHM datasets, particularly those for which natural frequencies are used as the damage-sensitive feature, one typically observes that the damage clusters radiate out from the healthy condition – often into the 'lower left' quadrant when damage amounts to a reduction in stiffness. In fact, this phenomenon can be seen in Figure 15 for the data generated in the current case study. In scenarios such as this, performing rotations around the centroid of the normal condition cluster can result in relatively-large displacements of the damage clusters, even for small angles of rotation. In unfavourable conditions, i.e. when principal directions of variation within the normal condition are not aligned with the relative directions of damage data, this will displace target-domain damage clusters far from the source-domain damage clusters. To mitigate the impact of this, when some damage data are available, it may be preferable to perform rotations around the centroid of all data. Of course, if damage data are unavailable, this will not be feasible and so one may seek to find an estimate for the centroid of all anticipated data via a physics-based model, such as a finite-element simulation.

The second important consideration is of the number of data points used to estimate class covariance matrices, which are in turn used to determine optimal rotations. If few data points are used to estimate the covariance matrices, there may be significant bias present which can in turn bias the rotations in the transfer mappings and lead to misalignment between source and target damage classes. The magnitude of this effect will be amplified by choosing a centre of rotation that is close to the boundary of the data.

A final point to consider is that, in general, scaling and rotation transformations do not commute. Therefore, one arrive at different transfer mappings depending on whether one opts to perform the rotation first and then scale (as in the proposed PCA-alignment variant), or scale first and then rotate (as in NCORAL). When performing scaling via standardisation, the rotation-scaling ordering leads to scaling along the principal directions

which may result in damage data being 'pushed away from' or 'pulled closer to' the normal condition data by greater amounts than if adopting the scaling-rotation ordering in which the data are scaled in the original feature basis. A systematic investigation into the conditions under which these different orderings are optimal is left as future work.

The rate of misclassification is of vital importance in SHM and PBSHM applications, as it directly impacts the performance in downstream decision-making tasks such as inspection scheduling and maintenance planning in a way that may incur unnecessary costs or undermine the safety of critical structures. For this reason, it is pertinent to understand, consider, and anticipate the effects and outcomes of any assumptions made when applying transfer-learning mappings in the context of PBSHM. The current subsection has highlighted some potential pitfalls when applying linear transformations learned from normal-condition data. Recent research has begun to address the drawbacks of learning mappings from the normal condition alone by adopting a sequential updating scheme for learning transfer mappings as more data become available [49]. Nonetheless, this research avenue remains relatively unexplored, and new solutions for learning transfer mappings under limited data are very much needed.

7. Discussion

The first general matter for discussion here is that of how rigorous the current picture is and exactly what the geometry has provided. Compared to the 'motivation' provided by the fibre-bundle picture in [7], the cellular sheaves provide an entirely rigorous formulation of feature spaces associated with structures. While this is useful on its own, the picture also provides a geometrical motivation for (linear) transfer via an adaption of the PCA-alignment algorithm of Singer and Wu [24]. In the manifold setting originally considered in [24], alignment corresponds to parallel transport and thus defines a covariant derivative and a corresponding connection. In the cellular sheaf picture, the view taken here is that alignment does correspond to parallel transport and thus motivates a definition of a discrete covariant derivative. The reason for this proposal is that graph theory provides a discrete derivative via the incidence matrix (as discussed in Section 3.1), and this may prove useful in subsequent analysis. The idea that one can define a *connection* on the cellular sheaf is more of a stretch.

There are a number of reasons for scepticism on the existence of a connection (in the usual sense of differential geometry [21]). A good reference on the construction of discrete connections is [50] (and the references therein). That paper is concerned with discrete connections for vector field analysis in computer graphics. The first major point of divergence with the analysis for a cellular sheaf is that the graph in the 'base space' is considered to be the triangulation of a manifold. That manifold is assumed to be a two-dimensional surface existing or embedded in $\mathbb{R}^3$. As a result of the embedding, the manifold inherits the Euclidean metric of the embedding. One imagines that the theory should generalise to a manifold of higher dimension embedded in some large Euclidean space. However, the situation for PBSHM is rather more complicated. The space of

structures in PBSHM is not readily interpreted as a triangulation of a manifold. The main problem is that one cannot think of the space of attributed graphs representing the structures as points of the *same dimension* (although a proposal for dealing with this will form part of the sister paper to this one). This simple fact means that the space of structures does not directly inherit a Euclidean metric from a straightforward embedding space. Although the space of structures does have 'metrics' – as discussed in Section 3.1 – they do not have the analytical properties required for a full implementation of the analysis used in [50]. The absence of a metric precludes the existence of a Levi-Civita (metric-compatible) connection [21]. Another problem there is that a Levi-Civita connection is associated with the tangent bundle of the base manifold and the 'fields'/features in the stalks of the cellular sheaf are *not* vector fields in the geometrical sense of sections of the tangent bundle. If one were to stretch the point, it is possible that the stalks of the cellular sheaf could be thought of as analogous to the fibres of an associated bundle, as in modern gauge theory[10] [51], to pursue parallel theory. In any case, for the purposes of this paper, a connection is not needed; the point is that cellular sheaves are a mathematically-rigorous picture of PBSHM in which theory-development can potentially take place. The first fruit of the picture is the emergence of linear transfer from the geometrical-alignment algorithm.

One might reasonably ask the question: why should one care to have a geometrical 'theory' of PBSHM? The most compelling answer to this question – in the opinion of the authors – is that an overarching theoretical framework allows the top-down design of systems implementing PBSHM, like the one proposed in [8]. PBSHM is not motivated by pure mathematics; however fanciful some engineers might consider some of the ideas here, the subject is motivated by the real need to develop practical methods of inferring the health of high-value structures without constant expensive inspections. The geometrical picture provides a solid basis for top-down design and provides a theoretical basis for the development of new PBSHM algorithms.

On the matter of new algorithms, the cellular sheaf framework has only provided linear transfer at the moment. To see the full power of domain adaption, one needs to consider the nonlinear embedding of the feature spaces in common latent spaces, as discussed in Section 4 and illustrated in Figure 9. In fact, this operation sits quite happily on top of the cellular sheaf picture, although the full DA approach does not yet have a direct geometrical motivation in that picture. Regardless, DA is very naturally accommodated in the cellular sheaf. In the viewpoint discussed in this paper so far, the stalks on the edges of the graphs have remained unused. In fact, the stalk on the edge between two structures S_1 and S_2 is a natural home for the latent space L of DA. Furthermore, as

[10] As discussed in [7], the operations of SHM data normalisation might be thought of as analogous to 'gauge-fixing' in the fibre-bundle picture of gauge theory. Although the current paper divests itself of the bundle picture, it would be interesting so see if data normalisation could have a geometrical interpretation in the cellular sheaf picture. Note that the GNN example discussed in Section 3.5 illustrated how a graph neural network could estimate a data-normalised zero section for a population of truss structures; given that sheaf neural networks generalise GNNs, one might hope for positive results on data normalisation.

stalks need not all have the same dimension, the idea of a reduced dimension for the latent features is also naturally accommodated.

Finally, it is worth mentioning an implementation issue in establishing a PBSHM framework. The linear transfer method induced by alignment is only truly appropriate to novelty detection; if it is to work for multi-class classification, one is faced with the strong assumption that any class boundaries built on source data will separate target data following the operations of alignment. This assumption may hold, if the 'distance' between structures is not too large, but it is then incumbent on the user not to damage important structure in the data by pre-processing. To give an example. Suppose one were to preprocess the data by 'sphering' the normal condition data; i.e., transforming the data into an isotropic Gaussian cluster. Clearly this operation would make the rotations needed in alignment problematical, as discussed in the last section. In practice, the calculations might lead to arbitrary rotations determined by fluctuations in the sampled data.

This latter thought raises an interesting point on feature selection; it suggests that 'isotropic' features could be problematic: are such features likely to occur? Consider the two types of vibrational features discussed in this paper so far: natural frequencies and frequency-response functions. In terms of natural frequencies, it is a commonly-held rule of thumb that these can be measured with 1% accuracy by modal testing. This rule indicates that the variance increases with frequency. A cluster of frequency features from multiple measurements will thus be an elliptical structure – centred on the 'truth', for unbiased estimators – with the major axes associated with the higher frequencies. The next observation one might make is that, if the major axes are aligned with the 'coordinate axes' in the feature space, then the rotation step will simply never be needed; however, again this seems unlikely. Consider a plate structure, where for the sake of argument, bending modes and torsional modes alternate. Now, suppose some overall mechanism determining bending stiffness and torsional stiffness. An increase in bending stiffness will increase the alternate bending frequencies with a corresponding effect for increases in torsional stiffness. This observation suggests that the frequencies will be correlated (in two groups), and this will rotate major axes of the normal condition cluster away from the coordinate axes. In fact, consideration of the analytical formulae for simple beams and plates [52] shows strong relations between natural frequencies. In terms of FRFs, one can appeal to the analysis in [53]. Assuming that the statistics for a given FRF value are Gaussian, the standard deviation for FRF magnitude $|H(\omega)|$ at a given frequency ω can be shown to be,

$$\sigma(|H(\omega)|) = \frac{\sqrt{1-\gamma^2(\omega)}}{|\gamma(\omega)|\sqrt{2N}}|H(\omega)| \tag{51}$$

where $\gamma(\omega)$ is the standard coherence function and N is the number of averages used to compute the FRF. This analysis shows that the noise in magnitude estimates is proportional to magnitude, so that one might expect magnitude features to be anisotropic.

Even so, there is the issue of the principal directions of the healthy-state data differing from the relative directions of damage clusters.

8. Conclusions

Population-based SHM so far, has been founded on an abstract representation theory of structures based on irreducible-element (IE) models and attributed graphs (AGs). Assuming this basis, the current paper constructs a mathematically-rigorous formulation of PBSHM, based on cellular sheaves, which very naturally incorporates transfer learning. In fact, the formulation provides a geometrical basis for a general (linear) form of transfer. It is argued here that such a mathematical formulation is a serious asset in terms of the design and implementation of practical PBSHM systems, in a top-down manner. The analysis builds on a previous geometrical picture – based on fibre bundles, which could only be regarded as an analogy – and provides rigour. A discrete version of parallel transport and a notion of covariant derivative are provided via a variant of the local PCA alignment algorithm of Singer and Wu as part of their development of vector diffusion maps (VDMs). The parallel analysis for VDMs at the cellular sheaf still requires development and will be a source of further work. Cellular sheaves are presented as the basis of sheaf neural networks (SNNs), which are generalised versions of graph neural networks, and their possible uses in PBSHM problems are discussed. The sheaf methodology developed here has been validated on a population of simulated structures and has shown excellent results in terms of novelty detection (unsupervised transfer learning) across populations. Problems in multi-class SHM problems have been discussed in terms of the rotation operation arising in the PCA-alignment operation; however, such problems can arise in any statistic-alignment algorithm that incorporates rotations. This problem will be considered further in future work.

As mentioned in the introduction, an alternative means of providing a geometrical PBSHM, would be to retain the fibre-bundle picture, but modify the space of structures in order that it have a manifold structure; this will be the subject of a sister paper, in which the idea of *parametric families* of structures will be introduced.

Acknowledgements

The authors would like to thank the EPSRC for funding from the EPSRC Programme Grants EP/R006768/1 (ROSEHIPS) and EP/Z531297/1 (CVD-NET). They would sincerely like to thank Dr Francesco Di Giovanni of Valence Labs for pointing them at the idea of sheaves on graphs.

For the purpose of open access, the authors have applied a Creative Commons Attribution (CC BY) licence to any Author Accepted Manuscript version arising.

References

[1] C.R. Farrar and K. Worden. *Structural Health Monitoring: a Machine Learning Perspective*. John Wiley & Sons, 2012.

[2] K. Worden, L.A. Bull, P.A. Gardner, J. Gosliga, T.J. Rogers, E.J. Cross, E. Papatheou, W. Lin, and N. Dervilis. An overview of recent developments in population-based structural health monitoring. *Frontiers in the Built Environment*, 6:Art.146, 2020.

[3] Q. Yang, Y. Zhang, W. Dai, and S.J. Pan. *Transfer Learning*. Cambridge University Press, 2020.

[4] L.A. Bull, P.A. Gardner, J. Gosliga, T.J. Rogers, N. Dervilis, E.J. Cross, E. Papatheou, A.E. Maguire, C. Campos, and K. Worden. Foundations of population-based SHM, Part I: Homogeneous populations and forms. *Mechanical Systems and Signal Processing*, 148:107141, 2021.

[5] J. Gosliga, P.A. Gardner, L.A. Bull, N. Dervilis, and K. Worden. Foundations of population-based structural health monitoring, part II: Heterogeneous populations – graphs, networks and communities. *Mechanical Systems and Signal Processing*, 148:107144, 2021.

[6] P.A. Gardner, L.A. Bull, J. Gosliga, N. Dervilis, and K. Worden. Foundations of population-based SHM, Part III: Heterogeneous populations – transfer and mapping. *Mechanical Systems and Signal Processing*, 149:107142, 2021.

[7] G. Tsialiamanis, C. Mylonas, E. Chatzi, N. Dervilis, D.J. Wagg, and K. Worden. Foundations of population-based SHM, Part IV: Geometry of spaces of structures and their feature spaces. *Mechanical Systems and Signal Processing*, 157:107692, 2021.

[8] D.S. Brennan, J. Gosliga, E.J. Cross, and K. Worden. Foundations of population-based SHM, Part V: Network, framework and database. *Mechanical Systems and Signal Processing*, 223:111602, 2025.

[9] S.J. Pan and Q. Yang. A survey on transfer learning. *IEEE Transactions on Knowledge and Data Engineering*, 22:1345–1359, 2010.

[10] L. Bull, P. Gardner, N. Dervilis, E. Papatheou, M. Haywood-Alexander, R. Mills, and K. Worden. On the transfer of damage detectors between structures: An experimental case study. *Journal of Sound and Vibration*, 501:116072, 2021.

[11] P. Gardner, L. Bull, N. Dervilis, and K. Worden. Overcoming the problem of repair in structural health monitoring: Metric-informed transfer learning. *Journal of Sound and Vibration*, 510:116245, 2021.

[12] F. Zhuang, Z. Qi, K. Duan, Y. Zhu, H. Xiong, and Q. He. A comprehensive survey on transfer learning. *Proceedings of the IEEE*, 109:43–76, 2021.

[13] Y. Gao and K.M. Mosalam. Deep transfer learning for image-based structural damage recognition. *Computer-Aided Civil and Infrastructure Engineering*, 33:748–768, 2018.

[14] S. Xu and H.Y. Noh. PhyMDAN physics-informed knowledge transfer between buildings for seismic damage diagnosis through adversarial learning. *Mechanical Systems and Signal Processing*, 151:107374, 2021.

[15] W. Zhang, G. Peng, C. Li, Y. Chen, and Z. Zhang. A new deep learning model for fault diagnosis with good anti-noise and domain adaptation ability on raw vibration signals. *Sensors*, 17, 2017.

[16] G. Michau and O. Fink. Domain adaptation for one-class classification: monitoring the health of critical systems under limited information. *International Journal of Prognostics and Health Management*, 10, 2019.

[17] X. Li, X.D. Jia, W. Zhang, H. Ma, Z. Luo, and X. Li. Intelligent cross-machine fault diagnosis approach with deep auto-encoder and domain adaptation. *Neurocomputing*, 383:235–247, 2020.

[18] Z. Chen, G. He, J. Li, Y. Liao, K. Gryllias, and W. Li. Domain adversarial transfer network for cross-domain fault diagnosis of rotary machinery. *IEEE Transactions on Instrumentation and Measurement*, 69:8702–8712, 2020.

[19] C.T. Wickramarachchi, A.E. Maguire, E.J. Cross, and K. Worden. Measuring data similarity in population-based structural health monitoring using distance metrics. *Structural Health Monitoring – An International Journal*, 23:2609–2635, 2024.

[20] N.E. Steenrod. *The Topology of Fibre Bundles*. Princeton University Press, 1992.

[21] S. Kobayashi and K. Nomizu. *Foundations of Differential Geometry*. Wiley, 2009.

[22] J. Hansen. A gentle introduction to sheaves on graphs. Technical report, https://www.jakobhansen.org/publications/gentleintroduction.pdf, 2021.

[23] R. Hartshorne. *Algebraic Geometry*. Springer Verlag, 1997.

[24] A. Singer and H.-T. Wu. Vector diffusion maps and the connection Lagrangian. *Communications in Pure and Applied Mathematics*, 65:1067–1144, 2012.

[25] D. Eisenbud and J. Harris. *Schemes: the Language of Modern Algebraic Geometry*. Wadsworth & Brookes/Cole, 1992.

[26] R. Diestel. *Graph Theory*. Springer-Verlag, New York, third edition, 2006.

[27] B. Bollobás. *Modern Graph Theory*. Springer, 1998.

[28] C. Bron and J. Kerbosch. Algorithm 457: finding all cliques of an undirected graph. *Communications of the ACM*, 16:575–577, 1973.

[29] P. Jaccard. Étude comparative de la distribution florale dans une portion des alpes et des jura. *Bulletin de la Société vaudoise des sciences naturelles*, 37:547–579, 1901.

[30] C.T. Wickramarachchi, J. Gosliga, A. Bunce, D.S. Brennan, D. Hester, E.J. Cross, and K. Worden. Similarity assessment of structures for population-based structural health monitoring via graph kernels. *Structural Health Monitoring – An International Journal*, 2024.

[31] A.J. Hughes, J. Poole, N. Dervilis, P.A. Gardner, and K. Worden. Quantifying the value of information transfer in population-based SHM. In *In Proceedings of the 42^{nd} International Modal Analysis Conference – IMAC-XLII, Orlando, FL*, 2024.

[32] J. Poole, P.A. Gardner, N. Dervilis, L.A. Bull, and K. Worden. On statistic alignment for domain adaptation in population-based structural health monitoring. *Structural Health Monitoring – An International Journal*, 22:1581—-1600, 2023.

[33] E.H. Spanier. *Algebraic Topology*. Springer-Verlag, 1994.

[34] G. Duthé, N. de Santos, I. Abdallah, W. Weijtjens, C. Devriendt, and E. Chatzi. Flexible multi-fidelity framework for load estimation of wind farms through graph neural networks and transfer learning. *Data-Centric Engineering*, 5:e29, 2024.

[35] X. Jian, K. Bacsa, G. Duthé, and E. Chatzi. Modal decomposition and identification for a population of structures using physics-informed graph neural networks and transformers. *Mechanical Systems and Signal Processing*, 241:113604, 2025.

[36] X. Jian, Y. Xia, G. Duthé, K. Bacsa, W. Liu, and E. Chatzi. Using graph neural networks and frequency domain data for automated operational modal analysis of populations of structures. *Data-Centric Engineering*, 6:e45, 2025.

[37] C. Bodnar, F. Di Giovanni, B.P. Chamberlain, P. Liò, and M. Bronstein. Neural sheaf diffusion: A topological perspective on heterophily and oversmoothing in GNNs. In *In Proceedings of the 36^{th} Conference on Neural Information Processing Systems (NeurIPS)*, 2022.

[38] F. Barbero, C. Bodnar, H.S̃aez de Ocáriz Borden, M. Bronstein, P. Velicković, and P. Liò. Sheaf neural networks with connection Laplacians. In *In Proceedings of the ICML 2022 Workshop on Topology, Algebra, and Geometry in Machine Learning*, 2022.

[39] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.H. Vetterling. *Numerical Recipes – The Art of Scientific Computing*. Cambridge University Press, 1986.

[40] G.H. Golub and C.F.Ṽan Loan. *Matrix Computations*. Johns Hopkins, 1996.

[41] N.J. Higham. Computing the polar decomposition with applications. *SIAM Journal of Scientific and Statistical Computing*, 7:1160–1174, 1986.

[42] C.M. Bishop. *Pattern Recognition and Machine Learning*. Springer, 2013.

[43] J. Zhang, W. Li, P. Ogunbona, and D. Xu. Recent advances in transfer learning for cross-dataset visual recognition: A problem oriented perspective. *ACM Computer Surveys*, 52:1–38, 2019.

[44] Q. Sun, R. Chattopadhyay, S. Panchanathan, and J. Ye. A two-stage weighting framework for multi-source domain adaptation. In *Advances in Neural Information Processing Systems*, volume 24, pages 1–9, 2011.

[45] B. Peeters and G.D̃e Roeck. One-year monitoring of the Z24-Bridge: environmental effects versus damage events. *Earthquake Engineering and Structural Dynamics*, 30:149–171, 2001.

[46] J. Poole, P. Gardner, A.J. Hughes, N. Dervilis, R.S. Mills, T.A. Dardeno, and K. Worden. Physics-informed transfer learning for SHM via feature selection. *Mechanical Systems and Signal Processing*, 237:113013, 2025.

[47] S. Christides and A.D.S. Barr. One-dimensional theory of cracked Bernoulli-Euler beams. *International Journal of Mechanical Sciences*, 26:639–648, 1984.

[48] K. Worden, T. Baldacchino J. Rowson, and E.J. Cross. Some recent developments in SHM based on nonstationary time series analysis. *Proceedings of the IEEE*, 104:1589–1603, 2016.

[49] J. Poole, N. Dervilis, K. Worden, P. Gardner, V. Giglioni, R.S. Mills, and A.J. Hughes. Active transfer learning for structural health monitoring. *Mechanical Systems and Signal Processing*, 241:113260, 2025.

[50] B. Liu, Y. Tong, F. De Goes, and M. Desbrun. Discrete connection and covariant derivative for vector field analysis and design. *ACM Transactions on Graphics*, 35:1–17, 2016.

[51] M. Nakahara. *Geometry, Topology and Physics*. Institute of Physics, 2003.

[52] R.D. Blevins. *Formulas for Natural Frequency and Mode Shape*. Krieger Publishing Company, 1979.

[53] J.S. Bendat and A.G. Piersol. *Engineering Applications of Correlation and Spectral Analysis*. John Wiley and Sons, 1980.